

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ДЕРЖАВНИЙ ЗАКЛАД  
«ЛУГАНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІМЕНІ ТАРАСА ШЕВЧЕНКА»

Навчально-науковий інститут фізики, математики  
та інформаційних технологій

Кафедра математики та інформатики

**Лазоренко Максим Олександрович**

**СИСТЕМА НАВЧАЛЬНОГО ПРИЗНАЧЕННЯ**

Магістерська робота  
за спеціальністю 014.09 «Середня освіта. Інформатика»

Особистий підпис -  \_\_\_\_\_

Науковий керівник -  д-р філософії В.Ю. Козуб

Зав. кафедри -  д.т.н., професор, Ю.Г. Козуб

Лубни - 2026

## АНОТАЦІЯ

**Лазоренко М.О.**

**Тема:** Система навчального призначення

**Спеціальність:** 014.09 „Середня освіта. Інформатика”

**Установа:** ДЗ ЛНУ імені Т.Шевченка, 2025 р.

**Магістерська робота містить:** 87 с., 26 рис., 1 табл., 1 додат. , 63 джерела.

**Об’єкт дослідження** - дистанційна освіта.

**Предмет дослідження** - архітектурні рішення, моделі даних та програмні засоби реалізації веб-систем навчального призначення.

**Мета роботи** - аналіз дистанційної освіти в Україні та світі і розробка системи навчального призначення

**Результати роботи:** В магістерській роботі здійснено аналіз сучасних підходів до розроблення систем навчального призначення, досліджено тенденції розвитку дистанційної освіти в Україні та світі, а також визначено основні проблеми й виклики, що виникають під час використання сучасних LMS у вітчизняних освітніх умовах. На основі проведеного аналізу сформовано вимоги до системи навчального призначення та спроектовано її архітектуру. Розроблено структурну модель бази даних і ER-діаграму, що забезпечує збереження навчального контенту, результатів навчальної діяльності та рольову модель доступу користувачів. Описано логіку взаємодії основних модулів системи, зокрема модулів керування користувачами, курсами, розділами, уроками, практичними завданнями та журналом оцінок.

**Висновок:** Розроблене програмне рішення відзначається структурованістю навчального контенту, зручністю використання та можливістю локального зберігання й перенесення навчальних матеріалів, що є особливо актуальним в умовах нестабільного доступу до мережі Інтернет. Запропонована система може бути використана в освітніх закладах, для індивідуального навчання або як основа для подальшого розвитку й розширення функціональних можливостей систем навчального призначення.

**Ключові слова:** БД, PHP, MySQL, Javascript, Laravel, ER-діаграму, LMS

## ABSTRACT

**Lazorenko M.**

**Subject:** Educational system.

**Specialty:** Secondary Education (Computer Science)

**Institution:** Luhansk University.

**Graduation work includes:** 87 pages, 26 images, 1 table, 1 attachment, 63 sources.

**Object of research:** remote education.

**Subject of research:** architectural solutions, data models, and software tools for implementing educational web systems.

**The aim of research:** to analyze remote education in Ukraine and worldwide and to develop an educational system.

**Results of the work:** The master's dissertation analyzes modern approaches to the development of educational systems, examines trends in the development of distance education in Ukraine and worldwide, and identifies the main problems and challenges that arise when using modern LMS in the domestic educational environment. Based on the analysis, requirements for the educational system were formulated and its architecture was designed. A structural model of the database and an ER diagram have been developed to ensure the preservation of educational content, learning outcomes, and a role-based user access model. The logic of interaction between the main modules of the system is described, in particular the modules for managing users, courses, sections, lessons, practical tasks, and the assessment log..

**Conclusion:** The developed software solution is characterized by the structured nature of the educational content, ease of use, and the ability to store and transfer educational materials locally, which is particularly relevant in conditions of unstable Internet access. The proposed system can be used in educational institutions, for individual learning, or as a basis for further development and expansion of the functional capabilities of educational systems.

**Keywords:** DB, PHP, MySQL, Javascript, Laravel, ER diagram, LMS.

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

БД - база даних;

ДНЗ - державний навчальний заклад;

ЕОМ - електронно-обчислювальна машина;

ІС - інформаційна система;

ІТ - інформаційні технології;

МВОК - масові відкриті дистанційні онлайн-курси;

НДІ - науково-дослідницький інститут;

ООП - об'єктно-орієнтоване програмування;

ОС - операційна система;

ПЗ - програмне забезпечення;

СУБД - система управління базами даних

LMS - learning management system

## ЗМІСТ

ВСТУП .....	7
РОЗДІЛ 1 АНАЛІЗ ДОСЛІДЖЕНЬ ТА СУЧАСНИХ РІШЕНЬ У ГАЛУЗІ СИСТЕМ НАВЧАЛЬНОГО ПРИЗНАЧЕННЯ.....	10
1.1. Поняття, роль та класифікація систем навчального призначення.....	10
1.2. Аналіз розвитку дистанційної освіти в Україні та світі.....	13
1.3. Огляд сучасних систем навчального призначення.....	16
1.4. Проблеми та виклики у розробленні сучасних систем навчального призначення. ....	22
РОЗДІЛ 2 АНАЛІЗ ТА ПРОЄКТУВАННЯ ВЕБ-СИСТЕМИ НАВЧАЛЬНОГО ПРИЗНАЧЕННЯ .....	28
2.1. Постановка задачі та загальна характеристика системи навчального призначення. ....	28
2.2. Функціональні вимоги до системи.....	30
2.3. Експлуатаційні вимоги та умови використання системи. ....	31
2.4 Клієнт - серверна архітектура.....	33
2.4.2. Сервер.....	34
2.4.3. Канали взаємодії та обмін даними .....	35
2.5. Серверна частина системи навчального призначення. ....	36
2.5.1. Використання мови програмування PHP .....	37
2.5.2. Застосування фреймворку Laravel .....	38
2.5.3. Використання системи керування базами даних MySQL .....	39
2.6. Архітектурна модель системи навчального призначення. ....	39
2.7. Моделювання структури бази даних системи навчального призначення. ....	42
2.8. Аналіз можливостей автономної (офлайн) роботи системи.....	44
РОЗДІЛ 3 ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ НАВЧАЛЬНОГО ПРИЗНАЧЕННЯ .....	48
3.1. Обґрунтування вибору засобів та технологій розроблення системи навчального призначення.....	48
3.2. Структура програмного проєкту системи навчального призначення ...	51

3.3. Реалізація взаємодії з базою даних - моделі та міграції .....	55
3.4. Реалізація основних модулів системи навчального призначення .....	58
3.5. Реалізація модуля користувачів .....	60
3.6. Реалізація модуля управління курсами та навчальним контентом .....	64
3.7. Реалізація модуля уроків і видів навчальної діяльності .....	67
3.8. Реалізація модуля журналу оцінок і результатів навчання .....	71
3.9 Структури бази даних для навчальних модулів .....	73
3.10. Реалізація механізму завантаження та перенесення курсу на пристрій користувача у вигляді архіву .....	77
ВИСНОВКИ.....	82
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	84

## ВСТУП

Сучасний етап розвитку освіти характеризується активною цифровізацією навчального процесу, зростанням ролі інформаційно-комунікаційних технологій та поширенням дистанційних і змішаних форм навчання. За цих умов особливого значення набувають системи навчального призначення, які забезпечують доступ до навчальних матеріалів, організацію навчальної діяльності, взаємодію між учасниками освітнього процесу та контроль результатів навчання. Вони стають невід'ємною складовою освітньої інфраструктури як у світовому, так і в українському контексті.

Актуальність теми дослідження зумовлена низкою чинників. По-перше, широке використання дистанційного навчання виявило низку проблем, пов'язаних із фрагментарністю цифрових інструментів, складністю наявних LMS, залежністю від стабільного доступу до мережі Інтернет та обмеженими можливостями довготривалого збереження навчальних матеріалів. По-друге, в умовах воєнного стану та частих технічних обмежень в Україні особливо гостро постає потреба у простих, надійних і структурованих програмних рішеннях, здатних забезпечувати безперервність навчального процесу та збереження освітнього контенту. По-третє, сучасні педагогічні підходи вимагають не лише подання навчального матеріалу, а й підтримки різних видів навчальної діяльності, фіксації результатів та надання зворотного зв'язку.

Незважаючи на наявність великої кількості готових платформ управління навчанням (Moodle, Google Classroom, Canvas та ін.), їх використання не завжди є оптимальним для закладів освіти з обмеженими технічними ресурсами або для індивідуального використання. Це зумовлює необхідність розроблення спеціалізованих веб-систем навчального призначення, орієнтованих на простоту використання, чітку структуру навчальних матеріалів, гнучкість архітектури та можливість локального збереження даних.

У зв'язку з цим актуальним є дослідження та розроблення системи навчального призначення як програмного засобу, що поєднує можливості сучасних веб-технологій із вимогами освітнього середовища та реальними умовами функціонування закладів освіти.

Метою магістерської роботи є проєктування та програмна реалізація системи навчального призначення, яка забезпечує структуроване зберігання навчальних матеріалів, підтримку навчальної діяльності користувачів та фіксацію результатів навчання в умовах дистанційної освіти.

Для досягнення поставленої мети в роботі необхідно розв'язати такі завдання:

- проаналізувати наукові підходи до визначення, ролі та класифікації систем навчального призначення;
- дослідити стан і тенденції розвитку дистанційної освіти в Україні та світі;
- виокремити основні проблеми та виклики, пов'язані з розробленням і використанням сучасних освітніх систем;
- сформулювати вимоги до системи навчального призначення з урахуванням умов її практичного використання;
- спроектувати архітектуру та структуру бази даних системи;
- реалізувати основні функціональні модулі системи з використанням сучасних веб-технологій;
- перевірити працездатність розробленої системи та оцінити можливості її практичного застосування.

Об'єктом дослідження є процес організації навчальної діяльності з використанням програмних засобів у дистанційному освітньому середовищі.

Предметом дослідження є методи, моделі та програмні засоби розроблення систем навчального призначення.

Методи дослідження, використані в роботі, включають аналіз і узагальнення наукових джерел, порівняльний аналіз існуючих систем, системний підхід до проєктування програмного забезпечення, моделювання структури даних та



архітектури системи, а також практичну реалізацію й апробацію розробленого програмного продукту.

Наукова новизна роботи полягає у проєктуванні системи навчального призначення, що поєднує класичну LMS-архітектуру з орієнтацією на структуроване збереження навчальних матеріалів, підтримку різних видів навчальної діяльності та можливість перенесення курсів у вигляді архівів для подальшого використання.

Практичне значення роботи полягає в можливості використання розробленої системи в освітніх закладах, для індивідуального навчання або як основи для подальшого розширення та адаптації під конкретні освітні потреби.

Загальний обсяг роботи – 80 сторінок; 63 джерела.

Структурно робота складається зі вступу, трьох розділів, висновків, списку використаних джерел та додатків. У першому розділі подано теоретичний аналіз проблематики та сучасного стану досліджень. Другий розділ присвячено проєктуванню системи та формуванню її алгоритмічної і структурної моделі. У третьому розділі описано програмну реалізацію системи навчального призначення та її основні функціональні модулі.

## **РОЗДІЛ 1**

### **АНАЛІЗ ДОСЛІДЖЕНЬ ТА СУЧАСНИХ РІШЕНЬ У ГАЛУЗІ СИСТЕМ НАВЧАЛЬНОГО ПРИЗНАЧЕННЯ**

#### **1.1. Поняття, роль та класифікація систем навчального призначення.**

У сучасній науково-педагогічній літературі поняття систем навчального призначення трактується багатозначно, що зумовлено різними підходами до розуміння сутності цифрових засобів навчання, їх функцій та місця в освітньому процесі. Узагальнюючи наявні підходи, можна стверджувати, що системи навчального призначення є складними програмно-інформаційними утвореннями, орієнтованими на підтримку навчальної діяльності.

Так, Т. А. Вакалюк визначає електронні засоби навчання як програмні засоби навчального призначення, у яких відображено певну предметну галузь та реалізовано технологію її вивчення із забезпеченням різних видів навчальної діяльності здобувачів освіти [1, с. 110-112]. Це визначення підкреслює дидактичну спрямованість таких засобів та їх роль у формуванні знань, умінь і навичок.

Системний підхід до трактування програмних засобів навчального призначення пропонує М. П. Шишкіна, яка розглядає їх як компоненти цілісних інформаційно-освітніх систем, призначених для підтримки діяльності зі знаннями на різних етапах навчального процесу - від подання та структурування інформації до її застосування і контролю результатів [2]. Такий підхід дозволяє перейти від розуміння окремих програм до осмислення систем навчального призначення як інтегрованих навчальних середовищ.

Подібну позицію обґрунтовує й В. Ю. Биков, який акцентує увагу на системності освітніх рішень та розглядає навчальні системи як складові організаційних систем відкритої освіти, що поєднують педагогічні, технологічні та управлінські компоненти [5, с. 45-52]. Це дає підстави

розглядати системи навчального призначення не лише як технічний інструмент, а як елемент освітньої інфраструктури.

Роль систем навчального призначення суттєво зростає в умовах цифровізації освіти та переходу до дистанційних і змішаних форм навчання. За результатами досліджень В. В. Галчанської, такі системи виконують низку важливих функцій: забезпечують доступ до навчальних матеріалів, підтримують комунікацію між учасниками освітнього процесу, дозволяють здійснювати контроль та оцінювання результатів навчальної діяльності, а також сприяють індивідуалізації навчання [3, с. 203-205].

Л. М. Гаврілова та Ю. О. Катасонова підкреслюють, що системи навчального призначення є ключовими компонентами систем дистанційного навчання, оскільки саме вони забезпечують інтерактивну взаємодію, зворотний зв'язок і доступ до навчального контенту незалежно від місця перебування здобувачів освіти [4, с. 172-173]. Таким чином, роль таких систем виходить за межі підтримки навчального процесу і охоплює організаційно-управлінські аспекти освіти.

Міжнародні дослідження також підтверджують зростання значення систем навчального призначення. Зокрема, у звітах UNESCO наголошується, що використання цифрових навчальних систем сприяє підвищенню доступності освіти, розвитку навичок самостійного навчання та формуванню цифрової компетентності [8]. Аналогічну позицію висловлює OECD, підкреслюючи роль таких систем у трансформації освітніх процесів у цифрову епоху [9].

Проблема класифікації систем навчального призначення залишається актуальною у зв'язку з різноманіттям їх функціональних можливостей та педагогічних завдань. Однією з найбільш обґрунтованих є діяльнісна класифікація, запропонована М. П. Шишкіною, відповідно до якої програмні засоби навчального призначення поділяються за типами діяльності, що

підтримуються: навчальні програми, тренажери, моделюючі середовища, системи контролю знань, експертні та інтелектуальні навчальні системи [2].

В. В. Галчанська доповнює зазначену класифікацію методичними критеріями, виділяючи навчальні, інформаційно-пошукові, контролюючі, імітаційні, демонстраційні та навчально-ігрові програмні засоби [3, с. 204-205]. Такий підхід є зручним для практичного використання у закладах освіти.

Сучасні дослідження також звертають увагу на еволюцію систем навчального призначення у напрямі інтелектуалізації. Так, О. В. Семеніхіна зазначає, що сучасні системи електронного навчання все частіше інтегрують елементи адаптивності, аналітики навчальних даних та персоналізації освітніх траєкторій [7, с. 18-25]. Це дозволяє говорити про поступовий перехід від традиційних програмних засобів до інтелектуальних навчальних систем.

Узагальнення вивчених підходів подано у Таблиці 1.

Таблиця 1

Основні підходи до класифікації систем навчального призначення

Автор	Критерій класифікації	Основні типи
Шишкіна М. П.	Типи діяльності зі знаннями	Навчальні програми, тренажери, моделюючі середовища, системи контролю, інтелектуальні системи
Галчанська В. В.	Методичне та функціональне призначення	Навчальні, контролюючі, інформаційно-пошукові, демонстраційні, імітаційні
Семеніхіна О. В.	Функціональні можливості	LMS, системи електронного навчання, адаптивні навчальні системи
Биков В. Ю.	Системний підхід	Компоненти освітніх інформаційних систем

## 1.2. Аналіз розвитку дистанційної освіти в Україні та світі.

Розвиток дистанційної освіти є закономірною відповіддю освітніх систем на глобальні соціальні, технологічні та економічні трансформації. У наукових дослідженнях наголошується, що дистанційне навчання перестало бути допоміжною формою і набуло статусу повноцінного елемента освітнього процесу, інтегрованого у національні та міжнародні стратегії розвитку освіти.

Становлення дистанційної освіти в Україні відбувалося поступово та супроводжувалося формуванням нормативно-правового підґрунтя і науково-методичного забезпечення. Як зазначають Л. М. Гаврілова та Ю. О. Катасонова, перші системні спроби впровадження дистанційного навчання були зумовлені потребою забезпечення доступу до освіти та модернізації традиційних форм навчання [10, с. 169-171]. Автори підкреслюють, що на початковому етапі дистанційне навчання розглядалося переважно як різновид заочного, однак із розвитком інформаційно-комунікаційних технологій воно трансформувалося у самостійну форму освітньої діяльності.

Вагомий внесок у теоретичне обґрунтування розвитку дистанційної освіти в Україні зробив В. Ю. Биков. У своїх працях він наголошує, що цифрові технології є ключовим чинником трансформації освітніх систем, а дистанційна освіта - складовою відкритої освіти, орієнтованої на доступність, гнучкість та індивідуалізацію навчання [11, с. 47-50]. Учений зазначає, що ефективність дистанційного навчання залежить від наявності цілісних інформаційно-освітніх систем, здатних забезпечувати управління навчальним процесом та збереження навчальних ресурсів.

Значну роль у розвитку дистанційної освіти відіграє педагогічний чинник. За результатами досліджень О. М. Спіріна, рівень цифрової компетентності педагогів безпосередньо впливає на якість організації дистанційного навчання [12, с. 14-18]. Автор підкреслює, що в українській практиці часто використовуються розрізнені цифрові інструменти, що ускладнює побудову цілісного освітнього середовища та знижує ефективність навчального процесу.

Суттєвого розвитку дистанційна освіта в Україні набула в умовах кризових викликів останніх років. С. Г. Литвинова зазначає, що вимушений перехід закладів освіти до дистанційного формату актуалізував проблеми нерівного доступу до цифрових ресурсів та нестачі структурованих навчальних матеріалів, придатних до тривалого використання [13, с. 5-8]. Дослідниця наголошує на потребі створення систем навчального призначення, здатних забезпечувати збереження та повторне використання навчального контенту.

Таким чином, розвиток дистанційної освіти в Україні характеризується нерівномірністю цифровізації, фрагментарністю використовуваних інструментів та зростаючою потребою у функціонально завершених системах навчального призначення, які можуть працювати в умовах обмеженого або нестабільного доступу до мережі Інтернет.

У світовому контексті дистанційна освіта розвивалася паралельно з удосконаленням інформаційних технологій та мережевої інфраструктури. У фундаментальній праці Handbook of Distance Education М. Мур зазначає, що дистанційне навчання формує нову педагогічну парадигму, у якій ключовими стають автономність здобувача освіти та опосередкована взаємодія через цифрові середовища [14, р. 3-7]. Автор наголошує, що сучасні дистанційні системи поєднують педагогічні, технологічні та організаційні компоненти.

Результати масштабного метааналізу, виконаного Б. Мінс та її колегами на замовлення Департаменту освіти США, свідчать, що за належної організації онлайн-навчання може забезпечувати не нижчі, а в окремих випадках вищі навчальні результати порівняно з традиційною формою [15, р. 9-11]. Дослідники підкреслюють, що ключову роль відіграє структура курсу, активна діяльність студентів та якість цифрових навчальних матеріалів.

Т. Андерсон у праці The Theory and Practice of Online Learning зазначає, що розвиток дистанційної освіти реалізує принцип «будь-де і будь-коли», знімаючи просторові та часові обмеження навчання [16, р. 41-44]. Автор

виділяє такі чинники ефективності онлайн-освіти, як доступність контенту, асинхронна взаємодія та персоналізація освітніх траєкторій.

Схожої позиції дотримується Д. Р. Гаррісон, який підкреслює, що електронне навчання є не альтернативою, а розвитком традиційних освітніх моделей [17, р. 28-31]. Він акцентує увагу на важливості когнітивної та соціальної присутності учасників освітнього процесу в цифровому середовищі.

Суттєвий вплив на глобальний розвиток дистанційної освіти справили масові відкриті онлайн-курси. За висновками Дж. Райха, MOOC сприяли розширенню доступу до освіти, однак їх ефективність значною мірою залежить від мотивації здобувачів та рівня інтерактивності курсів [18, р. 23-25].

Аналітичні звіти міжнародних організацій підтверджують зазначені тенденції. Зокрема, у доповіді UNESCO наголошується, що пандемія COVID-19 стала каталізатором глобального переходу до дистанційних форм навчання та прискорила цифрову трансформацію освіти [19, р. 6-9]. У звітах OECD підкреслюється зростання ролі цифрових платформ, аналітики навчальних даних та персоналізованого навчання як ключових напрямів подальшого розвитку дистанційної освіти [20, р. 33-36].

Отже, зарубіжні дослідження свідчать, що дистанційна освіта вже давно перестала бути допоміжною формою і стала повноцінною альтернативою традиційному навчанню. Її успішність визначається якістю цифрових інструментів, зокрема LMS та систем навчального призначення, а також рівнем структурованості й доступності навчальних ресурсів. Формується глобальний запит на системи, які поєднують мультимедійність, інтерактивність, мобільність і персоналізацію - саме такі характеристики визначають напрям розвитку сучасних освітніх платформ.

Таким чином, на відміну від глобальних центрів онлайн-освіти, де домінують великі хмарні платформи та аналітичні системи, український

контекст висуває інші пріоритети - простоту, автономність, структурованість та можливість тривалого локального зберігання навчальних матеріалів. Це створює об'єктивні передумови для розробки спеціалізованих веб-систем навчального призначення, які відповідають національним потребам і враховують особливості функціонування освіти в умовах воєнного стану та частих технічних обмежень.

### **1.3. Огляд сучасних систем навчального призначення.**

Сучасні системи навчального призначення (Learning Management Systems, LMS) є центральним компонентом цифрової освітньої інфраструктури. Вони забезпечують не лише доступ до навчального контенту, а й управління освітнім процесом, аналітику навчальних результатів, комунікацію між учасниками навчання та підтримку прийняття педагогічних і управлінських рішень [27, с. 4-7].

Аналітичні звіти OECD та UNESCO засвідчують формування глобальної тенденції до використання універсальних LMS-платформ як бази цифрової освіти. Водночас ефективність таких систем суттєво залежить від контексту їх застосування, технічної інфраструктури та рівня цифрових компетентностей користувачів [28, с. 22-27; 29, с. 15-19]

Для системного аналізу сучасних LMS у роботі використано такі критерії:

- функціональна повнота (курси, матеріали, тести, оцінювання);
- аналітика та звітність (прогрес, активність, результати);
- гнучкість і масштабованість;
- модель розгортання (хмара / локально / гібрид);
- автономність та офлайн-доступ;



- безпека та захист даних;
- придатність до українських умов (інтернет, навантаження, ресурси).

Зазначені критерії відповідають підходам, що застосовуються у сучасних порівняльних дослідженнях LMS-платформ [33, с.48-52; 21, с.3-6].

*Moodle* є найбільш поширеною LMS з відкритим програмним кодом, яка активно використовується в закладах освіти різних рівнів. Її ключовою перевагою є повна функціональна автономність - система може бути розгорнута локально на сервері закладу освіти або навіть на окремому комп'ютері [33, с. 131-134].

Аналітичні можливості Moodle включають детальний облік активності користувачів, результати тестування, логування дій та експорт даних. Це дозволяє здійснювати глибокий педагогічний аналіз, однак потребує належного налаштування і технічного супроводу [21, р. 6-9].

До основних недоліків системи дослідники відносять перевантажений інтерфейс для початківців, потребу в адмініструванні та складність адаптації під короткі або модульні курси [32, с. 115-118].

У контексті України Moodle є привабливою завдяки можливості локального зберігання навчальних матеріалів, що відповідає вимогам автономності та інформаційної безпеки [27, с. 10-12].

*Google Classroom* належить до хмарних систем навчального призначення і орієнтований на швидке впровадження та мінімальні технічні вимоги. Дослідження показують, що система ефективна для організації базового дистанційного навчання, особливо в кризових ситуаціях [31, с. 116-118].

До переваг Google Classroom належать інтуїтивний інтерфейс, інтеграція з Google Workspace та низький поріг входу для педагогів і здобувачів освіти. Водночас аналітичні огляди вказують на суттєві обмеження системи: відсутність повноцінної аналітики навчальних даних, слабку підтримку

складних структур курсів та повну залежність від постійного доступу до Інтернету [22, с. 45-47; 40, с. 7-9]

Таким чином, Google Classroom доцільний як тимчасове або допоміжне рішення, але не як довготривала система зберігання та управління навчальними матеріалами.

*Microsoft Teams* активно використовується як платформа синхронної взаємодії. Українські дослідження підтверджують ефективність Teams для організації онлайн-занять, консультацій і групових проєктів [32, с. 6-8].

Перевагами є:

- потужні засоби відеоконференцій;
- спільна робота з документами;
- інтеграція з Office 365.

Разом із тим, більшість дослідників наголошують, що Teams не є повноцінною LMS, оскільки:

- не забезпечує структурованого довготривалого зберігання навчальних курсів;
- має обмежені інструменти оцінювання;
- орієнтований радше на комунікацію, ніж на управління навчанням [21, с. 10-12; 33 с. 49-50].

*Canvas* - комерційна LMS, поширена у закладах вищої освіти США та Європи. Вона відзначається високим рівнем зручності користування, адаптивним дизайном і розвиненими інструментами аналітики [21, с. 7-9].

Аналіз джерел та практики використання вказує на такі переваги Canvas як сучасний інтерфейс, інтеграція з численними освітніми сервісами та масштабованість.

Недоліками системи є висока вартість ліцензування, орієнтація на хмарну модель, обмежені можливості локального розгортання [22, с. 48-50].

Для українських умов це знижує привабливість Canvas як базової платформи.

*Blackboard* є однією з найстаріших LMS-платформ і традиційно використовується у великих університетах. Вона пропонує розширені інструменти управління навчальним процесом, звітності та інтеграції з корпоративними системами [33, с.48-52].

Разом із тим, сучасні огляди відзначають складність інтерфейсу, високі вимоги до інфраструктури, високу вартість підтримки [21, с.8-10].

Ці фактори роблять *Blackboard* малопридатною для умов обмежених ресурсів.

Аналіз сучасних LMS-платформ у працях українських дослідників підтверджує, що ефективність систем навчального призначення визначається не лише набором функцій, а й здатністю підтримувати інтерактивну взаємодію та саморегуляцію навчальної діяльності. Дослідження, опубліковані у Віснику ЛНУ, демонструють, що в умовах змішаного та дистанційного навчання найбільш активно використовуються Moodle, Google Classroom і Microsoft Teams, однак їх педагогічний ефект є різним. Так, Moodle забезпечує найбільшу функціональну повноту та контроль навчальної діяльності, але потребує високого рівня цифрової компетентності педагогів і адміністративного супроводу [37, с. 118-120].

Google Classroom, навпаки, сприймається користувачами як просте та зручне середовище, проте обмежені можливості аналітики й відсутність гнучкої структури курсів знижують його придатність для довготривалого системного навчання [49, с. 160-162].

Microsoft Teams, за результатами українських досліджень, ефективний як комунікаційна платформа, але не забезпечує повноцінного управління навчальними курсами та архівування освітніх матеріалів [50, с. 29-31]. Це підтверджує висновок про те, що більшість популярних LMS орієнтовані або на управління контентом, або на комунікацію, але рідко поєднують обидва аспекти в єдиній автономній системі.

Однією з ключових тенденцій розвитку сучасних систем навчального призначення є інтеграція гейміфікаційних механік у навчальний процес. У наукових дослідженнях гейміфікація трактується як використання ігрових елементів (бали, рівні, досягнення, змагання, віртуальні нагороди) у неігровому контексті з метою підвищення мотивації, залученості та ефективності навчання.

Згідно з аналітичними звітами OECD, гейміфікація розглядається як інструмент персоналізації навчання та підтримки внутрішньої мотивації здобувачів освіти, особливо в умовах дистанційного та змішаного навчання [28, с. 45-49].

У класичних дослідженнях Deterding зазначається, що гейміфікація впливає насамперед на поведінкові та мотиваційні аспекти навчання, сприяючи регулярній активності користувачів у цифрових середовищах. Водночас автори наголошують, що ефективність гейміфікації залежить від її педагогічної доцільності та інтеграції в структуру навчального курсу, а не від механічного додавання ігрових елементів [42, с. 10-12].

У метааналізі досліджень гейміфікованого навчання автори доводять, що позитивний ефект спостерігається переважно у системах, де гейміфікація поєднується з чіткими цілями навчання та зворотним зв'язком, тоді як поверхнєве використання балів і рейтингів не дає сталих результатів [43, с. 161-164].

У сучасних LMS гейміфікація реалізується через:

- системи досягнень і бейджів;
- рівні прогресу та досвід (XP);
- сценарії місій і квестів;
- елементи соціальної взаємодії (команди, змагання).

Дослідження показують, що платформи з вбудованими гейміфікаційними модулями (наприклад, Moodle з плагінами, Classcraft,

Canvas із додатковими сервісами) демонструють вищий рівень активності користувачів порівняно з традиційними LMS [34, с. 422-428].

Разом із тим, більшість популярних LMS не мають глибоко інтегрованих гейміфікаційних механізмів на рівні архітектури, що змушує використовувати сторонні сервіси або плагіни. Це, у свою чергу, ускладнює адміністрування системи та знижує її автономність.

Українські дослідники активно аналізують потенціал гейміфікації в цифровій освіті. Так, у роботі Кременя, Бикова та Литвинової підкреслюється, що гейміфікація є складовою цифрової трансформації освіти та повинна розглядатися як елемент методичної підтримки, а не як самодостатній інструмент [27, с. 12-14].

Козуб Г. О., Шинкаренко Я. М. та Козуб В. Ю. на прикладі платформи Classcraft доводять, що впровадження гейміфікованих сценаріїв сприяє зростанню мотивації здобувачів освіти, підвищенню регулярності виконання завдань і формуванню навичок саморегуляції навчання [35, с. 20-24]. Автори наголошують, що ефективність гейміфікації значною мірою залежить від її інтеграції у загальну логіку навчального курсу.

У дослідженні Spaska A., Kozub H., Abylasynova G., Kozub V., Koval Y. зазначається, що використання сучасних інформаційних технологій у поєднанні з елементами гейміфікації позитивно впливає на результати навчання та сприяє розвитку цифрових компетентностей здобувачів освіти [34, р. 430-434].

Попри очевидні переваги, гейміфікація має низку обмежень. Зокрема, дослідники застерігають від надмірної ігровізації, яка може знижувати навчальну мотивацію та зміщувати акцент із досягнення освітніх цілей на формальне накопичення балів [43, с. 165-167]. Крім того, інтеграція гейміфікаційних механізмів потребує додаткових ресурсів, методичної підготовки педагогів і гнучкої архітектури системи навчального призначення.

Таким чином, гейміфікація є стійким глобальним трендом у розвитку сучасних систем навчального призначення. Вона демонструє значний потенціал для підвищення мотивації та залученості здобувачів освіти, однак потребує педагогічно обґрунтованої реалізації. В умовах України гейміфікація повинна поєднуватися з автономністю, простотою використання та можливістю локального зберігання навчальних матеріалів, що зумовлює необхідність розроблення спеціалізованих систем навчального призначення з вбудованими гейміфікаційними механізмами.

#### **1.4. Проблеми та виклики у розробленні сучасних систем навчального призначення.**

Активний розвиток дистанційної та змішаної освіти, особливо у контексті досвіду пандемії, безпекової ситуації та блекаутів безумовно мають вплив на освітній процес, що супроводжується низкою проблем і викликів, що мають як технологічний, так і педагогічний, організаційний та соціальний характер. Аналіз наукових досліджень українських і зарубіжних авторів свідчить, що наявні системи не завжди відповідають реальним умовам функціонування освіти, особливо в кризових і нестабільних середовищах [27, с. 12-15; 28, с. 30-34].

Однією з ключових проблем сучасних систем навчального призначення є їх висока залежність від стабільного доступу до мережі Інтернет. Більшість популярних LMS (Google Classroom, Microsoft Teams, Canvas, Blackboard) орієнтовані на хмарну модель розгортання, що унеможливує повноцінну роботу системи в умовах обмеженого або нестабільного з'єднання [22, с. 45-47; 33, с. 48-50].

Українські дослідники підкреслюють, що перебої з інтернет-з'єднанням та електропостачанням негативно впливають на безперервність навчального процесу, регулярність виконання завдань і збереження результатів навчальної

діяльності [37, с. 121-123]. У таких умовах особливої актуальності набуває питання автономності систем навчального призначення, тобто можливості локального зберігання навчальних матеріалів і результатів роботи користувачів.

Додатковим технологічним викликом є інформаційна безпека та захист персональних даних. Вимоги міжнародного стандарту ISO/IEC 27001:2022 передбачають комплексний підхід до управління інформаційною безпекою, який у багатьох освітніх платформах реалізується частково або фрагментарно [38, с. 12-18]. Це створює ризики витоку даних та обмежує довіру користувачів до цифрових освітніх систем.

Також, необхідно зважати на педагогічні виклики пов'язані насамперед із обмеженою інтерактивністю та формалізацією навчального процесу у більшості LMS. Морозова М. В. наголошує, що відсутність автономного режиму роботи LMS унеможливорює підтримку інтерактивності та безперервності навчального процесу [48, с. 129-131]. У свою чергу, Караман О. Л. підкреслює, що цифрові системи повинні забезпечувати локальне збереження навчальних матеріалів і результатів діяльності, що є важливою умовою формування цифрової компетентності учасників освітнього процесу [50, с. 30-32].

Дослідження українських учених свідчать, що значна частина навчальних курсів у LMS зводиться до розміщення текстових матеріалів і завдань без належного зворотного зв'язку та адаптації до індивідуальних потреб учнів [49, с. 160-162]. Це суперечить сучасним підходам до цифрової освіти, орієнтованим на активне, персоналізоване та компетентнісне навчання.

Окремої уваги потребує гейміфікація навчального процесу, яка розглядається як перспективний засіб підвищення мотивації. Водночас наукові дослідження вказують, що поверхнєве впровадження ігрових елементів без педагогічного обґрунтування не дає сталого позитивного ефекту

і може призводити до зниження навчальної спрямованості [43, р. 165-167; 35, с. 20-24].

Важливою проблемою є недостатній рівень цифрової компетентності педагогів, що ускладнює ефективне використання можливостей систем навчального призначення. Як зазначає О. М. Спірін, педагоги часто використовують лише базовий функціонал платформ, не задіюючи інструменти аналітики, адаптації та управління навчальним процесом [10, с. 17].

Рамка цифрової компетентності педагогічних працівників визначає необхідність системного розвитку навичок роботи з цифровими освітніми середовищами, однак на практиці ці вимоги не завжди реалізуються через обмежені ресурси та відсутність єдиного підходу до вибору і впровадження LMS [37, с. 6-7].

Аналітичні звіти UNESCO та OECD підкреслюють, що цифрова трансформація освіти супроводжується зростанням цифрової нерівності, залежністю від комерційних платформ і ризиком втрати контролю над освітніми даними [18, с. 6-9; 28, с. 30-34]. В українських реаліях ці проблеми посилюються умовами воєнного стану, що вимагає створення стійких, гнучких і автономних освітніх систем.

Отже, аналіз проблем і викликів у розробленні сучасних систем навчального призначення дозволяє виокремити такі ключові напрями, що потребують подальшого дослідження та вдосконалення:

- забезпечення автономності та стійкості систем;
- підвищення рівня інтерактивності та педагогічної доцільності цифрових курсів;
- інтеграція гейміфікації на основі науково обґрунтованих підходів;
- розвиток цифрових компетентностей педагогів;
- гарантування безпеки та захисту освітніх даних.



Зазначені проблеми обґрунтовують необхідність розроблення спеціалізованої системи навчального призначення, орієнтованої на автономну роботу, структуроване збереження навчальних матеріалів та адаптацію до реальних умов функціонування освіти.

Таким чином, аналіз сучасних викликів дає змогу сформулювати ключові критерії, яким повинні відповідати перспективні системи навчального призначення:

1. Простота та інтуїтивність інтерфейсу, мінімізація зайвих функцій.
2. Підтримка автономної роботи та локального зберігання даних.
3. Стандартизованість структур навчальних матеріалів.
4. Гнучкість та розширюваність архітектури системи.
5. Захищеність персональних та навчальних даних.
6. Адаптивність та персоналізація освітнього процесу.
7. Можливість швидкого розгортання без значних технічних ресурсів.

У сукупності ці виклики визначають потребу у створенні програмного продукту нового типу - легкої, автономної та структурно гнучкої веб-системи навчального призначення, здатної працювати у різних технічних умовах та адаптуватися під потреби українських закладів освіти. Саме такі висновки закладають основу для подальшого алгоритмічного та програмного проєктування системи, що розглядається у наступних розділах роботи.

## Висновки до розділу 1

У першому розділі здійснено теоретико-аналітичний огляд сучасного стану досліджень у сфері систем навчального призначення та дистанційної освіти. Узагальнення наукових підходів дало змогу уточнити зміст поняття «система навчального призначення» як комплексного програмно-інформаційного утворення, орієнтованого на підтримку навчальної діяльності, організацію взаємодії учасників освітнього процесу, керування навчальними ресурсами та контроль результатів навчання. Показано, що такі системи слід розглядати не лише як окремі програмні засоби, а як складові цілісних інформаційно-освітніх середовищ, у яких поєднуються педагогічні, технологічні та організаційні компоненти.

Проведений аналіз класифікацій засвідчив різноманітність підходів до типологізації систем навчального призначення: за видами діяльності зі знаннями, методичним призначенням, функціональними можливостями та системним підходом. Встановлено, що еволюція таких систем відбувається у напрямі інтеграції елементів адаптивності, персоналізації та аналітики навчальних даних, що відображає глобальний тренд переходу від окремих навчальних програм до комплексних цифрових освітніх платформ.

У підрозділі, присвяченому розвитку дистанційної освіти, визначено, що дистанційне навчання у світі трансформувалося у повноцінний компонент освітніх систем, ефективність якого значною мірою залежить від якості цифрових інструментів, структури курсів і організації навчальної діяльності. Водночас український контекст характеризується нерівномірністю цифровізації, фрагментарністю застосовуваних сервісів та зростаючою потребою у функціонально завершених системах, здатних забезпечити структурованість і довготривале збереження навчальних матеріалів.

Аналіз проблем і викликів розроблення сучасних систем навчального призначення дозволив виокремити ключові обмеження наявних LMS: складність і перевантаженість інтерфейсів, залежність від мережевої

інфраструктури, недостатню стандартизацію навчальних матеріалів, UX-проблеми, ризики безпеки персональних даних та обмежену адаптивність навчальних середовищ. Зазначені чинники формують вимоги до перспективних освітніх систем: простота, зрозуміла структура контенту, надійність зберігання матеріалів, гнучка архітектура, захищеність даних, можливість розширення та адаптації під потреби користувачів.

Таким чином, результати першого розділу підтверджують актуальність розроблення веб-орієнтованої системи навчального призначення, яка забезпечує структуроване подання навчальних матеріалів, підтримку навчальної діяльності та фіксацію результатів, а також відповідає вимогам українського освітнього середовища. Отримані висновки слугують теоретичним підґрунтям для формування вимог і проєктних рішень, що розглядаються у наступних розділах роботи.

## РОЗДІЛ 2

### АНАЛІЗ ТА ПРОЄКТУВАННЯ ВЕБ-СИСТЕМИ НАВЧАЛЬНОГО ПРИЗНАЧЕННЯ

#### **2.1. Постановка задачі та загальна характеристика системи навчального призначення.**

На підставі аналізу сучасного стану дистанційної освіти та існуючих систем навчального призначення, виконаного у першому розділі, можна зробити висновок про наявність суперечності між зростаючими вимогами до організації навчального процесу та обмеженими можливостями наявних програмних рішень. З одного боку, сучасна освіта потребує гнучких, адаптивних і зручних інформаційних систем, а з іншого - більшість поширених платформ є або надмірно складними, або критично залежними від хмарної інфраструктури та стабільного інтернет-з'єднання.

У зв'язку з цим виникає необхідність у створенні системи навчального призначення, яка б забезпечувала ефективну підтримку освітнього процесу в умовах дистанційного або змішаного навчання та дозволяла здійснювати довготривале зберігання навчальних матеріалів у структурованому вигляді на локальному комп'ютері користувача. Така система повинна поєднувати переваги сучасних веб-технологій із простотою використання та автономністю роботи.

Об'єктом дослідження у межах даного розділу є веб-орієнтована система навчального призначення як програмно-інформаційний засіб підтримки навчального процесу. Предметом дослідження є алгоритмічні, структурні та архітектурні рішення, що забезпечують функціонування системи в автономному та гібридному режимах.

З позицій системного аналізу система навчального призначення розглядається як складний програмно-інформаційний об'єкт, що функціонує у взаємодії з користувачем та зовнішнім середовищем. Основними елементами

цієї системи є інформаційні ресурси, програмні модулі, інтерфейс користувача та апаратне середовище, в якому здійснюється експлуатація. Кожен з перелічених елементів впливає на загальну ефективність навчального процесу, а порушення функціонування будь-якого з них призводить до зниження якості навчання.

У межах даного дослідження система навчального призначення розглядається з позицій методу системного аналізу, що дозволяє представити її як цілісну систему, яка складається з взаємопов'язаних компонентів: програмної архітектури, інформаційних ресурсів, інтерфейсу користувача та механізмів керування даними. Застосування цього методу дає змогу комплексно оцінити вплив кожного з компонентів на ефективність функціонування системи в цілому.

Аналіз існуючих систем навчального призначення дозволяє виокремити їхні позитивні сторони, зокрема широкий функціонал, підтримку різних форматів навчального контенту та інтеграцію з мережевими сервісами. Водночас встановлено низку недоліків, серед яких найбільш суттєвими є залежність від постійного доступу до мережі Інтернет, складність локального розгортання та підвищені вимоги до апаратних ресурсів користувача. Причинами зазначених недоліків є орієнтація більшості сучасних LMS на централізовану хмарну інфраструктуру та використання складних програмних рішень, що не завжди є доцільними для умов обмежених ресурсів.

Основною задачею розробки є створення програмного засобу, який забезпечує роботу з навчальними матеріалами незалежно від наявності постійного доступу до мережі Інтернет, підтримує логічну ієрархію курсів і тем, а також реалізує механізми контролю знань. Система орієнтована на використання в закладах загальної середньої та вищої освіти, а також у межах індивідуального навчання.

## **2.2. Функціональні вимоги до системи.**

Формування функціональних вимог здійснювалося з використанням методу функціонального аналізу, який передбачає визначення ключових функцій системи та оцінку їхнього впливу на результативність навчального процесу. Застосування цього методу дозволило встановити пріоритетність функцій, безпосередньо пов'язаних із поданням навчального контенту, контролем знань та збереженням результатів навчальної діяльності.

З метою поглибленого дослідження можливостей системи доцільно згрупувати її функції за призначенням. До навчальних функцій належать подання та структуризація навчального контенту, забезпечення навігації між темами та підтримка різних форматів матеріалів. Контрольні функції охоплюють організацію тестування, автоматизоване оцінювання та фіксацію результатів навчальної діяльності. Сервісні функції включають керування користувачами, збереження індивідуального прогресу та адміністрування системи.

Застосування функціонального аналізу дозволяє встановити, що критичними для забезпечення безперервності навчального процесу є саме навчальні та контрольні функції, тоді як сервісні функції мають допоміжний характер. Такий розподіл дає змогу оптимізувати ресурси системи та зосередити увагу на реалізації найбільш значущих компонентів.

Порівняння з типовими LMS показує, що значна частина функціоналу таких систем є надмірною для індивідуального або локального навчання, що ускладнює інтерфейс і підвищує вимоги до технічних ресурсів. Запропонований підхід передбачає реалізацію лише функцій, які безпосередньо впливають на якість навчання.

Проектована система повинна забезпечувати створення, перегляд і впорядкування навчальних курсів, які мають ієрархічну структуру та складаються з тем і окремих навчальних матеріалів. Навчальні матеріали

можуть бути представлені у вигляді текстових файлів, PDF-документів, зображень або мультимедійних ресурсів. Такий підхід забезпечує гнучкість використання системи в межах різних освітніх дисциплін.

Важливою функцією є організація тестування для контролю знань. Система повинна підтримувати створення тестів різних типів, автоматичне оцінювання результатів та збереження історії проходження тестів. Реалізація цієї функції дозволяє здійснювати об'єктивний контроль рівня засвоєння навчального матеріалу та аналізувати навчальний прогрес користувачів у динаміці.

Окрему увагу приділено функціям керування користувачами. Система повинна забезпечувати реєстрацію, авторизацію, збереження персональних даних і розмежування прав доступу. Наявність ролі адміністратора дозволяє здійснювати централізоване керування навчальним контентом і параметрами системи.

Таким чином, сформульовані функціональні вимоги визначають систему як комплексний інструмент, здатний забезпечити повний цикл роботи з навчальним матеріалом - від подання інформації до оцінювання результатів навчання.

### **2.3. Експлуатаційні вимоги та умови використання системи.**

Експлуатаційні вимоги визначають умови, за яких програмний засіб має коректно функціонувати та бути придатним до практичного застосування в освітньому процесі. Проектована система навчального призначення розробляється як веб-застосунок, що працює у типовому середовищі користувача з використанням сучасного веб-браузера та серверної частини на PHP із застосуванням фреймворку Laravel і СКБД MySQL.

Система повинна забезпечувати стабільну роботу під час виконання основних сценаріїв користувача: перегляд структури курсів, доступ до

навчальних матеріалів, проходження тестування та збереження результатів. З метою забезпечення практичної придатності програмного продукту передбачається використання поширених технологій і стандартних інструментів розгортання, що спрощує встановлення та подальшу експлуатацію системи в закладах освіти або в індивідуальному використанні.

Важливою умовою є відповідність системи принципам зручності та доступності інтерфейсу. Інтерфейс користувача повинен бути логічно структурованим, інтуїтивно зрозумілим і не перевантаженим елементами, що сприяє ефективному використанню системи користувачами з різним рівнем цифрової компетентності.

Для забезпечення цілісності даних та коректності роботи система повинна підтримувати механізми валідації введених даних, обробки помилок та журналювання ключових подій. Крім того, система має бути спроектована таким чином, щоб її функціонал можна було розширювати без суттєвої перебудови архітектури, що забезпечує перспективність подальшого розвитку програмного продукту.

Разом з тим, аналіз експлуатаційних характеристик показує, що використання веб-технологій у локальному середовищі може супроводжуватися певними обмеженнями, зокрема необхідністю попереднього налаштування серверного програмного забезпечення та бази даних. Причиною цього є орієнтація сучасних веб-фреймворків переважно на серверне або хмарне розгортання. Проте зазначені недоліки мають технічний характер і можуть бути мінімізовані шляхом використання стандартних інструментів локального розгортання.

Аналіз показує, що в умовах використання персональних комп'ютерів з обмеженими ресурсами критично важливою є оптимізація програмного коду та мінімізація вимог до апаратного забезпечення. Саме це обумовлює вибір веб-орієнтованого підходу з можливістю локального розгортання системи без залучення складної серверної інфраструктури.



## 2.4 Клієнт - серверна архітектура.

Архітектура клієнт-сервер є базовою моделлю побудови сучасних веб-застосунків і забезпечує розподіл функцій між двома основними сторонами: клієнтською (браузер користувача) та серверною (веб-сервер і прикладна логіка). Для системи навчального призначення така архітектура є доцільною, оскільки дозволяє централізовано керувати навчальними даними, забезпечувати рольовий доступ до функціоналу та підтримувати одночасну роботу багатьох користувачів.[53]

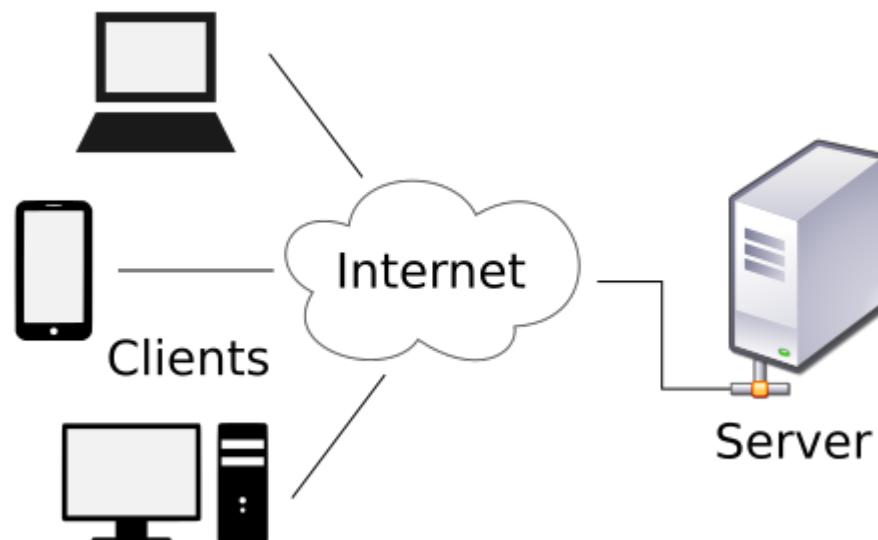


Рис. 2.1 Ілюстрація клієнт-серверної архітектури

У розробленій системі клієнтська частина відповідає за відображення інтерфейсу користувача, взаємодію з навчальним контентом, введення даних (реєстрація, авторизація, виконання завдань) та ініціювання запитів до сервера. Серверна частина виконує обробку запитів, реалізує бізнес-логіку, здійснює перевірку прав доступу, працює з базою даних і повертає результати у вигляді сформованих веб-сторінок або структурованих відповідей.

### 2.4.1. Клієнт

Клієнтська частина функціонує у веб-браузері та реалізується з використанням стандартних веб-технологій HTML, CSS і JavaScript. Основним завданням клієнта є забезпечення зручного доступу до функцій системи для різних категорій користувачів (вчитель і студент). Інтерфейс розробляється таким чином, щоб мінімізувати когнітивне навантаження, забезпечити зрозумілу навігацію між курсами, темами й уроками, а також надати засоби взаємодії з навчальними матеріалами і результатами.

Клієнтська частина також відповідає за виконання базових перевірок введених даних (наприклад, обов'язковість заповнення полів), зручність заповнення форм, візуальне відображення статусів (успішні дії, помилки), а також динамічні елементи інтерфейсу, такі як модальні вікна створення тем і уроків.[55]

#### **2.4.2. Сервер**

Серверна частина системи реалізує прикладну логіку та функції керування даними. Вона приймає HTTP-запити від клієнта, проводить їхню обробку, виконує перевірку автентичності користувача та відповідності його прав доступу до запитуваного ресурсу. Серверна частина забезпечує виконання основних сценаріїв: управління курсами, додавання тем і уроків, завантаження навчальних матеріалів, створення завдань, приймання відповідей, збереження результатів та формування журналу оцінок.

Важливою функцією сервера є централізований контроль доступу до інформації. Для цього у системі реалізується рольова модель доступу, відповідно до якої вчитель має права на створення і редагування навчального контенту, а студент - права на перегляд матеріалів та виконання завдань. Серверна логіка гарантує, що критичні операції (наприклад, редагування курсів або виставлення оцінок) не можуть бути виконані користувачем без відповідних прав.[63]

### 2.4.3. Канали взаємодії та обмін даними

Взаємодія клієнта й сервера відбувається за протоколом HTTP/HTTPS. У відповідь на запити клієнта сервер формує сторінки інтерфейсу або повертає дані, необхідні для оновлення окремих елементів сторінки. Для підвищення зручності користування та зменшення кількості повних перезавантажень сторінок можуть використовуватися асинхронні запити (AJAX), зокрема під час додавання елементів контенту або перевірки даних.

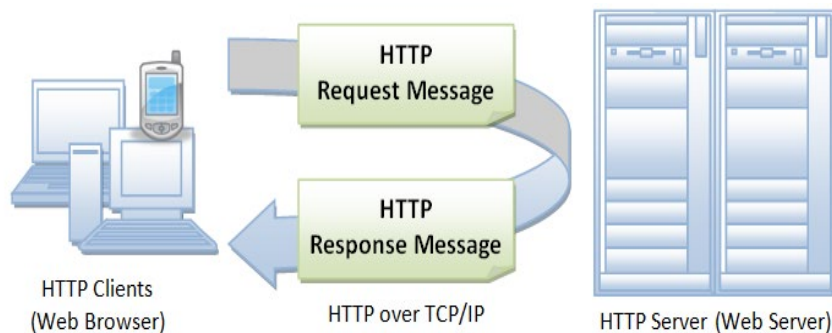


Рис. 2.2 Ілюстрація схеми запит - відповідь

Окрему категорію становить робота з файлами навчальних матеріалів. У цьому випадку клієнт ініціює завантаження файлів на сервер, після чого сервер виконує валідацію, зберігає файл у файловому сховищі та фіксує метадані (назва, тип, прив'язка до уроку) у базі даних. Такий підхід забезпечує цілісність навчальних матеріалів і зручність їх використання в межах курсу.

### 2.4.4. Переваги клієнт-серверної архітектури

Використання клієнт-серверної архітектури забезпечує низку практично важливих переваг:

- Централізоване керування даними. Навчальні матеріали, результати й права доступу зберігаються та контролюються на сервері, що підвищує узгодженість даних.
- Безпека та контроль доступу. Сервер виконує перевірку прав користувача на кожну операцію, що зменшує ризик несанкціонованих дій.

- Масштабованість. Архітектура дозволяє обслуговувати зростаючу кількість користувачів без зміни клієнтської частини.
- Зручність оновлення. Оновлення серверної частини не потребує перевстановлення застосунку на пристроях користувачів.
- Кросплатформеність. Користувачі працюють із системою через браузер незалежно від операційної системи та типу пристрою. [52]

Таким чином, клієнт-серверна архітектура є обґрунтованою основою для реалізації системи навчального призначення, оскільки забезпечує розподіл функцій, централізований контроль та зручність використання в освітньому процесі. У наступних підрозділах буде деталізовано вибір та особливості реалізації серверної частини системи на основі технологій PHP, Laravel і MySQL.

## **2.5. Серверна частина системи навчального призначення.**

Серверна частина системи навчального призначення відповідає за обробку запитів користувачів, реалізацію бізнес-логіки, взаємодію з базою даних та забезпечення цілісності й безпеки навчальних даних. Вибір технологій серверної частини здійснювався з урахуванням вимог до надійності, масштабованості, простоти супроводу та можливості подальшого розвитку системи [63]

Для реалізації серверної частини у роботі обрано зв'язку **PHP - Laravel - MySQL**, яка є поширеною у веб-розробці, має добре документовану архітектуру та відповідає потребам створення систем навчального призначення.

### 2.5.1. Використання мови програмування РНР

РНР є однією з найбільш розповсюджених серверних мов програмування для розроблення веб-застосунків. Вибір РНР як базової мови серверної частини зумовлений її орієнтованістю на веб-середовище, підтримкою об'єктно-орієнтованого програмування та широкими можливостями інтеграції з різними системами керування базами даних.

Суттєвою перевагою РНР є простота розгортання та сумісність з більшістю веб-серверів і операційних систем. Це дозволяє використовувати розроблену систему як у локальному середовищі, так і на віддалених серверах без значних змін конфігурації. Для освітніх закладів з обмеженими технічними ресурсами така універсальність є важливою практичною перевагою.[57]

РНР також має велику екосистему бібліотек і спільноту розробників, що сприяє стабільності та довготривалій підтримці проєктів. Для систем навчального призначення це означає можливість швидкої реалізації типових функцій, таких як авторизація користувачів, робота з файлами навчальних матеріалів, обробка форм і керування сесіями.

Таким чином, використання РНР забезпечує надійну основу для реалізації серверної логіки системи навчального призначення та створює умови для її подальшого вдосконалення.[57]



Рис. 2.3 Лого серверної мови програмування РНР

### 2.5.2. Застосування фреймворку Laravel

Для підвищення структурованості та якості програмного коду у роботі використано фреймворк **Laravel**, який є одним із найпопулярніших фреймворків для мови PHP. Laravel реалізує архітектурну модель MVC (Model-View-Controller), що дозволяє чітко розмежувати рівень даних, бізнес-логіку та інтерфейс користувача. [55, 56]

Однією з ключових переваг Laravel є наявність вбудованих механізмів маршрутизації, автентифікації та авторизації, що значно спрощує реалізацію систем із рольовим доступом. Це є особливо актуальним для систем навчального призначення, у яких необхідно чітко розмежовувати функціональні можливості вчителів і студентів.

Фреймворк надає інструменти для роботи з базою даних через ORM Eloquent, що дозволяє оперувати даними на рівні об'єктів, а не безпосередніх SQL-запитів. Крім того, механізм міграцій забезпечує керування версіями структури бази даних, що підвищує стабільність і відтворюваність системи під час розгортання.

Laravel також підтримує модульний підхід до побудови застосунків, що сприяє розширюваності системи. Це дозволяє додавати нові функціональні модулі (наприклад, аналітику навчальної діяльності або імпорт/експорт курсів) без суттєвих змін у базовій архітектурі.



Рис 2.4 Лого фреймворку Laravel

### 2.5.3. Використання системи керування базами даних MySQL

Для зберігання навчальної інформації в системі використовується реляційна система керування базами даних **MySQL**, яка є стабільною, продуктивною та добре інтегрується з мовою PHP і фреймворком Laravel. Реляційна модель даних є доцільною для систем навчального призначення, оскільки навчальний контент і результати навчання мають чітко визначену структуру та логічні зв'язки.

MySQL забезпечує підтримку транзакцій, індексів і механізмів забезпечення цілісності даних, що є важливим для збереження інформації про курси, уроки, подання робіт та оцінки. Завдяки цьому система може гарантувати узгодженість даних навіть у разі одночасної роботи кількох користувачів.

Ще однією перевагою MySQL є її поширеність і невисокі вимоги до ресурсів, що дозволяє використовувати систему як на серверних платформах, так і в локальних інсталяціях. Це відповідає вимогам освітніх закладів, де часто використовується різноманітне апаратне забезпечення.

У поєднанні з ORM Eloquent MySQL забезпечує ефективну взаємодію з серверною логікою системи, спрощує виконання запитів і підвищує читабельність програмного коду.[58]



Рис 2.5 Лого СКБД MySQL

## **2.6. Архітектурна модель системи навчального призначення.**

Проектування архітектури системи здійснювалося із застосуванням методу структуризації, який є конкретизацією системного аналізу та дозволяє розподілити складний програмний продукт на логічно завершені функціональні компоненти.

У процесі проектування архітектури системи було розглянуто декілька альтернативних підходів, зокрема монолітну архітектуру та архітектуру односторінкових застосунків (SPA). Монолітний підхід характеризується простотою реалізації, проте ускладнює масштабування та супровід програмного продукту. Архітектура SPA забезпечує високу інтерактивність, однак потребує додаткових клієнтських ресурсів і складнішої реалізації.

Порівняльний аналіз показав, що архітектурна модель MVC є найбільш доцільною для системи навчального призначення, оскільки забезпечує чітке розмежування відповідальності між компонентами, зручність модифікації та підтримку модульності. Це дозволяє адаптувати систему до різних сценаріїв використання без суттєвих змін базової архітектури.

Проектування архітектури системи навчального призначення є важливим етапом алгоритмічного конструювання, оскільки саме архітектурні рішення визначають логіку взаємодії компонентів, масштабованість та зручність супроводу програмного продукту. Для реалізації поставлених у роботі задач доцільно застосувати архітектурну модель MVC (Model-View-Controller), яка є стандартною для сучасних веб-застосунків і повністю підтримується фреймворком Laravel.

Згідно з даною моделлю, модель (Model) відповідає за роботу з даними та реалізацію бізнес-логіки системи. Вона забезпечує взаємодію з базою даних MySQL, обробку навчальної інформації, збереження результатів тестування та управління об'єктами предметної області, такими як курси, теми, матеріали та користувачі.

Контролер (Controller) виконує функцію посередника між моделлю та інтерфейсом користувача. Він обробляє запити, що надходять від клієнтської



частини, ініціює виконання відповідних алгоритмів, здійснює перевірку прав доступу та передає оброблені дані до компонентів відображення. Такий підхід дозволяє централізувати керування логікою роботи системи та спростити її модифікацію.

Подання (View) відповідає за візуальне представлення інформації та взаємодію з користувачем. Реалізація подання здійснюється за допомогою шаблонів Blade, а також мов HTML, CSS і JavaScript, що забезпечує динамічне та зручне відображення навчального контенту у веб-браузері.

Архітектура системи передбачає поділ функціоналу на окремі модулі, серед яких можна виокремити модуль управління користувачами, модуль навчальних курсів, модуль тестування та модуль зберігання навчальних матеріалів. Такий модульний підхід сприяє підвищенню читабельності коду, зменшенню залежностей між компонентами та полегшує подальше розширення системи.

Застосування архітектурної моделі MVC у поєднанні з можливостями фреймворку Laravel дозволяє створити структурований, масштабований та зручний у супроводі програмний продукт, що відповідає вимогам сучасних систем навчального призначення та створює надійне підґрунтя для реалізації програмної частини, описаної у наступному розділі.

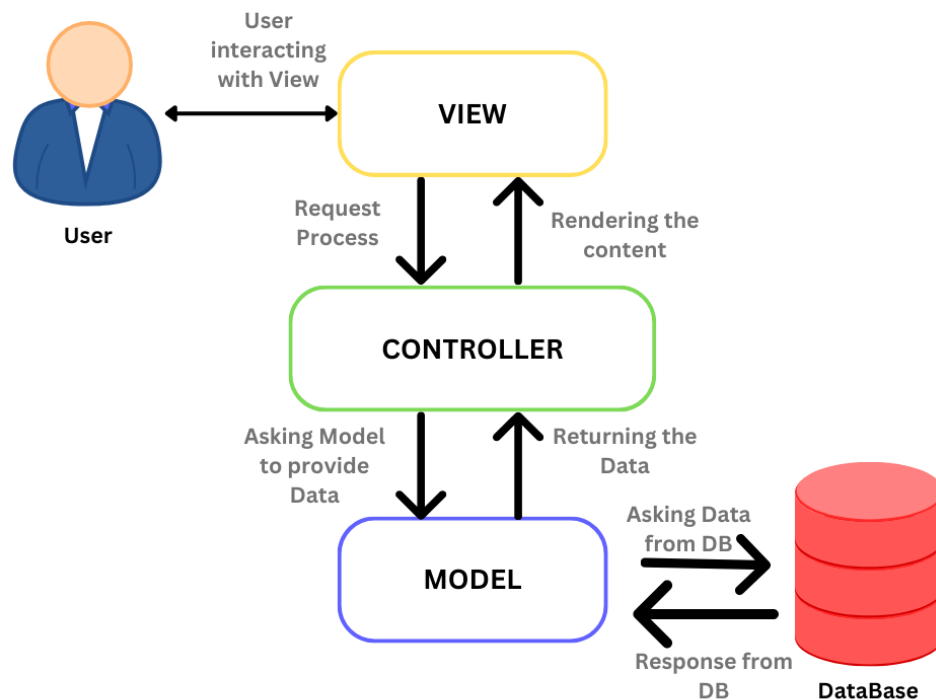


Рис. 2.6 Ілюстрація схеми MVC.

## 2.7. Моделювання структури бази даних системи навчального призначення.

Ефективне функціонування системи навчального призначення значною мірою залежить від коректно спроектованої структури бази даних, яка забезпечує зберігання, обробку та цілісність навчальної інформації. З огляду на характер задач, що вирішуються у межах даної роботи, доцільним є використання реляційної системи керування базами даних MySQL, яка широко застосовується у веб-застосунках та повністю інтегрується з фреймворком Laravel.

Проектування структури бази даних здійснюється з урахуванням принципів нормалізації, що дозволяє зменшити дублювання даних і

підвищити ефективність виконання запитів. Логічна модель бази даних будується на основі аналізу предметної області та відображає основні сутності системи навчального призначення, а також зв'язки між ними.

До основних сутностей бази даних належать користувачі, навчальні курси, теми, навчальні матеріали, тести, запитання тестів та результати тестування. Сутність «користувач» містить інформацію, необхідну для ідентифікації та авторизації, а також для збереження результатів навчальної діяльності. Сутність «курс» використовується для групування навчального матеріалу за певною дисципліною або напрямом підготовки.

Навчальні курси мають ієрархічну структуру та поділяються на окремі теми. Кожна тема може містити один або декілька навчальних матеріалів, представлених у різних форматах. Такий підхід забезпечує логічну організацію контенту та зручність навігації для користувачів системи.

Для реалізації механізмів контролю знань у базі даних передбачено окремі сутності для тестів та запитань. Кожен тест пов'язаний з певною темою або курсом і містить набір запитань, які використовуються для оцінювання рівня засвоєння навчального матеріалу. Результати тестування зберігаються в окремій таблиці, що дозволяє відстежувати навчальний прогрес користувачів у динаміці.

Між основними сутностями встановлюються зв'язки типу «один до багатьох», що відповідає логіці організації навчального процесу. Наприклад, один курс може містити декілька тем, одна тема - декілька навчальних матеріалів або тестів, а один користувач може мати багато записів результатів тестування. Така модель забезпечує цілісність даних та спрощує реалізацію алгоритмів роботи системи.

Запропонована структура бази даних є достатньо гнучкою для подальшого розширення функціоналу системи, зокрема додавання нових типів навчальних матеріалів або розширення механізмів оцінювання. Крім того, використання стандартних можливостей MySQL у поєднанні з інструментами

міграцій Laravel дозволяє спростити процес супроводу та модифікації бази даних у процесі розвитку програмного продукту.

Вибір реляційної СКБД MySQL обґрунтований її надійністю, широкою підтримкою у веб-середовищі та можливістю ефективної роботи в локальному режимі. Крім того, використання механізмів міграцій дозволяє забезпечити контроль версій структури бази даних та спрощує процес супроводу системи.

Таким чином, моделювання структури бази даних створює надійне інформаційне підґрунтя для реалізації системи навчального призначення та забезпечує ефективну взаємодію між програмними модулями, що буде детально розглянуто у наступних підрозділах і в третьому розділі роботи.

## **2.8. Аналіз можливостей автономної (офлайн) роботи системи.**

В умовах нестабільного доступу до мережі Інтернет, зокрема в регіонах з обмеженою інфраструктурою або під час надзвичайних ситуацій, особливої актуальності набуває можливість автономної (офлайн) роботи систем навчального призначення. Автономний режим передбачає функціонування програмного продукту без постійного підключення до зовнішніх мереж, із збереженням основних освітніх функцій та доступом до навчального контенту.[61, 62]

Більшість сучасних систем управління навчанням (LMS), таких як Moodle або Google Classroom, орієнтовані насамперед на роботу в онлайн-середовищі. Їхня функціональність суттєво залежить від наявності Інтернет-з'єднання, оскільки навчальні матеріали, завдання, результати оцінювання та комунікація між учасниками освітнього процесу зберігаються та обробляються на віддалених серверах. У разі втрати доступу до мережі користувачі втрачають можливість повноцінної роботи з такими системами.

Разом з тим, автономні або частково автономні системи навчального призначення мають низку важливих переваг. По-перше, вони забезпечують

безперервність навчального процесу незалежно від зовнішніх технічних факторів. Навчальні матеріали (тексти лекцій, презентації, відеофайли, тести) можуть зберігатися локально на комп'ютері користувача у структурованому вигляді, що дозволяє працювати з ними в будь-який час. По-друге, офлайн-доступ сприяє підвищенню надійності збереження даних, оскільки користувач не залежить від доступності віддаленого сервера або хмарного сервісу.

З технічної точки зору автономна робота системи може реалізовуватися шляхом використання локальних баз даних (наприклад, SQLite або локального MySQL-сервера), файлової системи для зберігання навчального контенту та локального веб-сервера. Такий підхід дозволяє створити програмний продукт, який за своєю логікою нагадує вебзастосунок, але функціонує повністю або частково в межах одного комп'ютера. Для користувача це означає збереження звичного інтерфейсу та принципів взаємодії без необхідності постійного підключення до Інтернету.

Важливим аспектом автономної роботи є можливість синхронізації даних. У разі відновлення Інтернет-з'єднання система може передбачати механізми експорту та імпорту результатів навчання, оновлення контенту або резервного копіювання даних. Такий гібридний підхід поєднує переваги офлайн- та онлайн-режимів, забезпечуючи гнучкість і масштабованість навчального процесу.

Серед недоліків автономних систем слід відзначити складність централізованого адміністрування та оновлення програмного забезпечення, а також обмежені можливості колективної роботи в реальному часі. Проте для індивідуального навчання, довготривалого збереження навчальних матеріалів та використання в умовах обмеженого доступу до мережі ці недоліки є менш критичними. Таким чином, аналіз можливостей автономної (офлайн) роботи системи навчального призначення свідчить про її високу практичну цінність. Реалізація офлайн-функціональності дозволяє забезпечити стійкість навчального процесу, доступність освітніх ресурсів та незалежність

користувача від зовнішніх інфраструктурних чинників. Це робить автономні системи перспективним напрямом розвитку програмних засобів для освіти, особливо в умовах дистанційного та змішаного навчання.

## Висновки до розділу 2

У другому розділі здійснено комплексний аналіз об'єкта та предмета дослідження з використанням методів системного, функціонального та технічного аналізу. Надано загальну характеристику системи навчального призначення, визначено її основні функціональні та експлуатаційні вимоги, а також обґрунтовано вибір архітектурної моделі та структури бази даних.

Проведений аналіз існуючих систем навчального призначення дозволив виявити їхні позитивні сторони, зокрема функціональну насиченість та підтримку сучасних форматів навчального контенту, а також встановити низку суттєвих недоліків, пов'язаних із залежністю від інтернет-з'єднання та складністю локального використання. Причини зазначених недоліків мають переважно технологічний характер і обумовлені орієнтацією більшості LMS на хмарну інфраструктуру.

Особливу увагу приділено аналізу можливостей автономної роботи системи, що дозволило обґрунтувати доцільність розробки програмного засобу, орієнтованого на офлайн- або гібридний режим використання. З огляду на результати аналізу визначено основні принципи проєктування системи, які лягли в основу подальшої програмної реалізації.

Отримані у другому розділі результати створюють теоретичне та технічне підґрунтя для розробки програмної реалізації системи навчального призначення, що буде детально розглянуто у третьому розділі кваліфікаційної роботи.

## **РОЗДІЛ 3**

### **ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ НАВЧАЛЬНОГО ПРИЗНАЧЕННЯ**

#### **3.1. Обґрунтування вибору засобів та технологій розроблення системи навчального призначення**

Програмна реалізація системи навчального призначення потребує використання надійних, поширених і технологічно зрілих засобів розроблення, які дозволяють створити функціональний, зручний у використанні та придатний до супроводу програмний продукт. Вибір програмних технологій здійснювався з урахуванням вимог, сформульованих у другому розділі роботи, а також особливостей освітнього середовища, в якому передбачається використання системи.

Однією з ключових вимог до програмного забезпечення є можливість створення веб-орієнтованої системи з чіткою структурою, підтримкою роботи з базами даних, механізмами авторизації користувачів і зручним інтерфейсом. З огляду на це як основну серверну мову програмування було обрано PHP, яка є однією з найбільш поширених мов веб-розробки у світі. PHP характеризується простотою синтаксису, широкими можливостями для реалізації серверної логіки та високим рівнем сумісності з різними операційними системами та веб-серверами.

Важливою перевагою PHP є її активна спільнота та наявність великої кількості бібліотек і фреймворків, що значно прискорюють процес розроблення. Для реалізації системи навчального призначення це дозволяє зосередитися на логіці роботи з навчальним контентом і користувачами, не витрачаючи значні ресурси на реалізацію базових механізмів, таких як маршрутизація запитів, обробка форм або робота з сесіями.

З метою підвищення якості програмного коду та забезпечення його структурованості у роботі використано фреймворк Laravel, який є одним з



найпопулярніших PHP-фреймворків. Laravel реалізує архітектурну модель MVC (Model-View-Controller), що дозволяє логічно розмежувати рівень даних, бізнес-логіку та інтерфейс користувача. Такий підхід підвищує зрозумілість коду, спрощує його супровід та дає можливість розширювати функціонал системи без суттєвих змін існуючої структури.

Laravel також надає вбудовані засоби для реалізації механізмів автентифікації та авторизації користувачів, що є критично важливим для систем навчального призначення. Використання готових рішень зменшує ймовірність помилок, пов'язаних із безпекою, та дозволяє забезпечити контроль доступу до навчальних матеріалів і результатів тестування. Крім того, фреймворк підтримує механізм міграцій бази даних, який забезпечує контроль версій структури даних і спрощує процес розгортання системи.

Для зберігання інформації про користувачів, курси, теми, навчальні матеріали та результати тестування у роботі використано систему керування базами даних MySQL. Вибір MySQL зумовлений її стабільністю, високою продуктивністю та широким використанням у веб-застосунках. MySQL добре інтегрується з Laravel через ORM Eloquent, що дозволяє працювати з базою даних на рівні об'єктів і зменшує складність реалізації запитів.

Використання реляційної бази даних є доцільним з огляду на чітко визначену структуру навчальної інформації та наявність логічних зв'язків між сутностями, такими як курси, теми, матеріали та результати навчання. Це дозволяє забезпечити цілісність даних, уникнути дублювання інформації та ефективно реалізувати алгоритми роботи системи.

Клієнтська частина системи реалізується з використанням стандартних веб-технологій - HTML, CSS та JavaScript, що забезпечує доступність системи у будь-якому сучасному веб-браузері без потреби встановлення додаткового програмного забезпечення. Мова HTML використовується для структуризації вмісту сторінок, CSS - для формування зовнішнього вигляду інтерфейсу, а

JavaScript - для реалізації інтерактивних елементів і підвищення зручності взаємодії користувача з системою.

Для формування динамічних сторінок застосовується шаблонізатор Blade, який є складовою фреймворку Laravel. Використання Blade дозволяє ефективно поєднувати серверну логіку з елементами інтерфейсу, забезпечує повторне використання шаблонів і спрощує підтримку єдиного стилю оформлення системи.

Обраний стек технологій також забезпечує можливість подальшого розвитку програмного продукту. Зокрема, система може бути розширена шляхом додавання нових модулів, інтеграції додаткових засобів аналізу навчального прогресу або вдосконалення інтерфейсу користувача без необхідності повного перепроєктування. Це є важливою характеристикою для програмних систем навчального призначення, які повинні адаптуватися до змін освітніх вимог і технологічних умов.

Таким чином, використання PHP у поєднанні з фреймворком Laravel, системою керування базами даних MySQL та стандартними веб-технологіями HTML, CSS і JavaScript є обґрунтованим і доцільним вибором для реалізації системи навчального призначення. Обрані засоби розроблення забезпечують необхідний рівень функціональності, надійності та зручності використання, а також створюють технічне підґрунтя для подальшої програмної реалізації, яка буде детально розглянута у наступних підрозділах третього розділу.

### **3.2. Структура програмного проєкту системи навчального призначення**

Структура програмного проєкту системи навчального призначення визначає організацію програмного коду, розподіл відповідальностей між компонентами та логіку взаємодії між ними. Коректно спроектована структура проєкту є важливою умовою забезпечення зрозумілості, супровідності та можливості подальшого розвитку програмного продукту.

У межах даної роботи структура програмного проєкту сформована відповідно до архітектурної моделі MVC, яка реалізується фреймворком Laravel. Такий підхід передбачає логічне розмежування програмного коду на три основні складові: моделі, контролери та подання, що сприяє підвищенню якості реалізації та спрощує підтримку системи.

Основні компоненти програмного проєкту зосереджені у стандартних каталогах фреймворку Laravel. Каталог `app` містить основну серверну логіку системи. У ньому зосереджені моделі, які відповідають за роботу з даними, а також контролери, що реалізують алгоритми взаємодії користувача з системою навчального призначення. Моделі відображають сутності предметної області, зокрема користувачів, навчальні курси, теми, матеріали та результати тестування, і забезпечують доступ до відповідних таблиць бази даних.

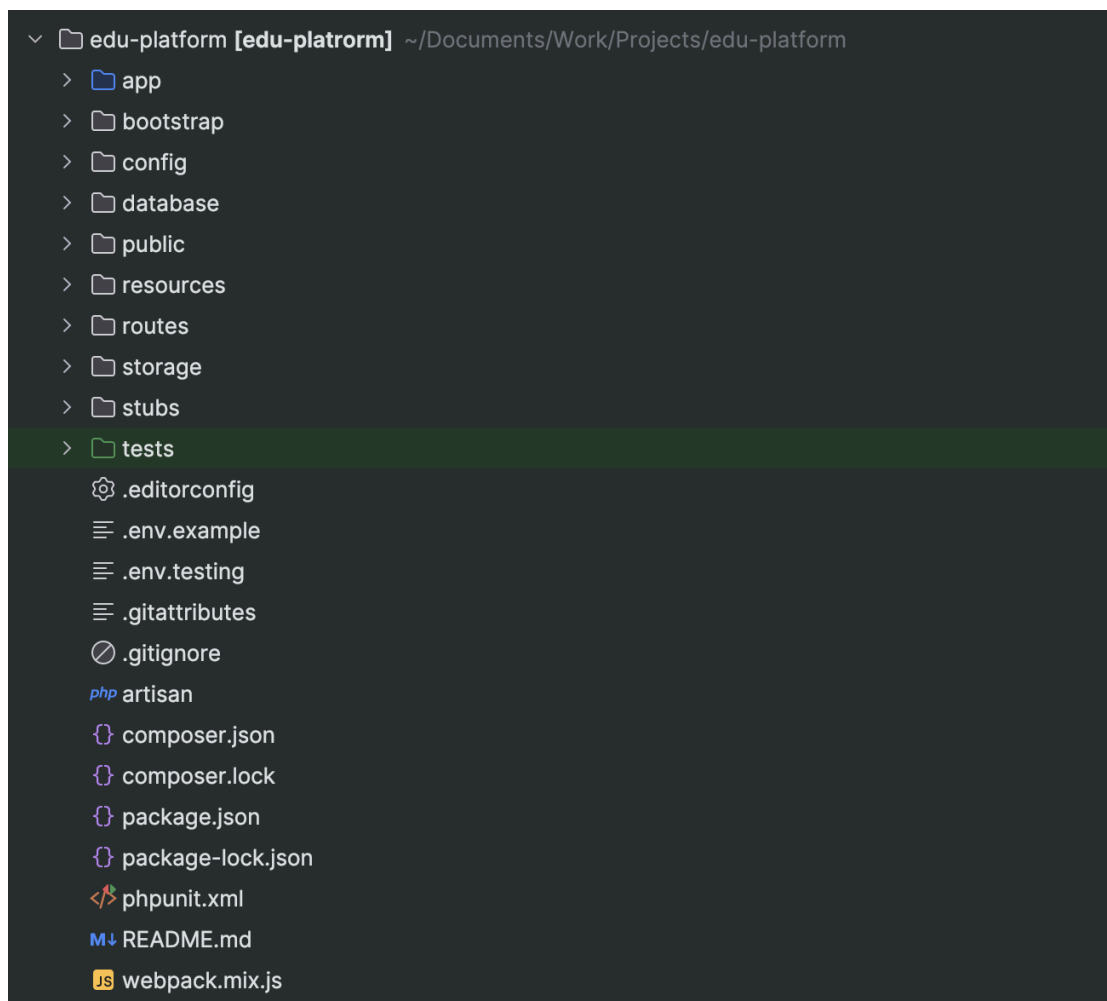


Рис. 3.1 Структура проєкту на фреймворці Laravel

Контролери системи відповідають за обробку HTTP-запитів, виконання бізнес-логіки та передачу даних до інтерфейсу користувача. Для зручності супроводу та логічної організації контролери групуються відповідно до функціональних модулів системи, таких як управління користувачами, навчальними курсами та тестуванням. Така організація сприяє зменшенню зв'язності між компонентами та підвищує читабельність програмного коду.

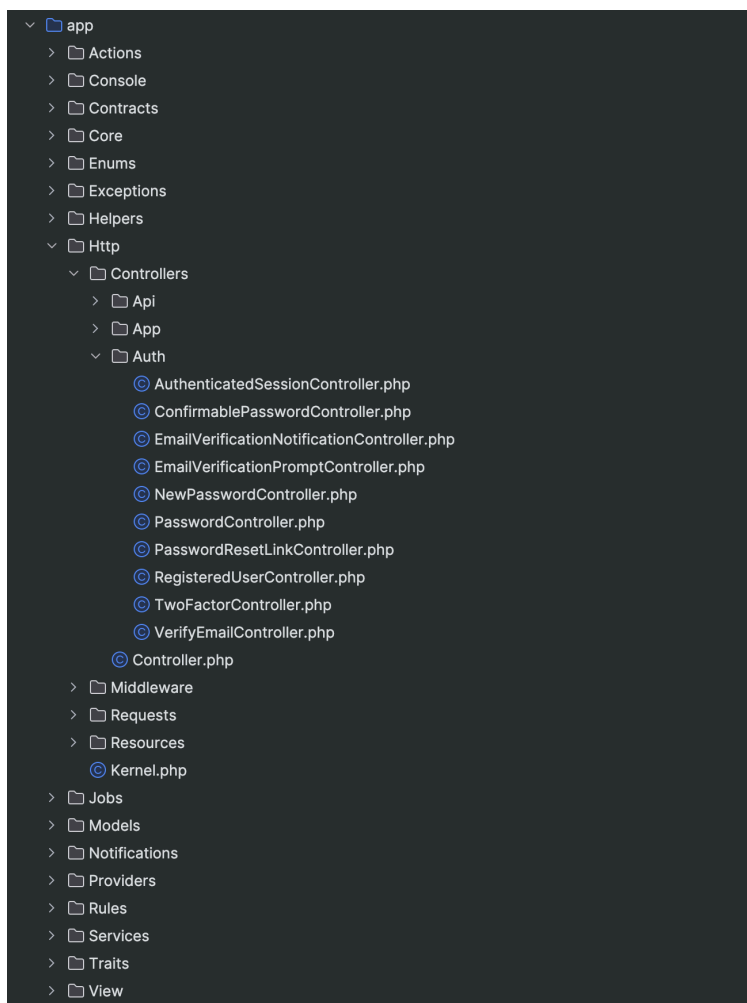


Рис. 3.2 Директорія app та піддиректорія Http/Controllers

Каталог `resources` містить файли, що відповідають за інтерфейс користувача. У підкаталозі `views` розміщуються шаблони Blade, які визначають структуру та зовнішній вигляд веб-сторінок системи. Шаблони поділяються на базові компоненти та сторінки окремих функціональних модулів, що забезпечує повторне використання елементів інтерфейсу та єдність стилю оформлення. У цьому ж каталозі зосереджені ресурси стилів і скриптів, що використовуються для формування візуальної складової системи.



Рис. 3.3 Структура директорії resources

Маршрутизація запитів у системі реалізується за допомогою механізмів Laravel і визначається у відповідних файлах маршрутизації. Маршрути забезпечують зв'язок між URL-адресами та відповідними контролерами, що дозволяє централізовано керувати логікою доступу до функціональних можливостей системи. Такий підхід спрощує контроль прав доступу та забезпечує зрозумілу навігацію в межах застосунку.



Рис. 3.4 Структура директорії routes

Окреме місце у структурі проєкту займає каталог database, який містить файли міграцій та початкового наповнення бази даних. Використання міграцій дозволяє формалізувати процес створення та зміни структури бази даних, забезпечує контроль версій і полегшує перенесення проєкту між різними середовищами. Це є важливим чинником стабільності та відтворюваності програмного продукту.

Файлова структура системи організована таким чином, щоб навчальні матеріали зберігалися окремо від програмного коду, що підвищує безпеку та спрощує управління контентом. Такий підхід дозволяє оновлювати або доповнювати навчальні ресурси без внесення змін до логіки роботи системи.

Загалом, обрана структура програмного проєкту відповідає сучасним вимогам до веб-застосунків, забезпечує чіткий поділ відповідальностей між компонентами та створює умови для подальшого розширення функціоналу системи навчального призначення. Вона також слугує надійним підґрунтям для реалізації основних модулів системи, які детально розглядаються у наступних підрозділах третього розділу.

### 3.3. Реалізація взаємодії з базою даних - моделі та міграції

Реалізація взаємодії з базою даних є фундаментальним етапом програмної реалізації системи навчального призначення, оскільки саме на цьому рівні відбувається збереження навчальної інформації, управління зв'язками між сутностями та забезпечення цілісності даних. У межах даної роботи база даних використовується для зберігання відомостей про користувачів, навчальні курси, теми, матеріали, тести та результати навчальної діяльності.

Для реалізації доступу до бази даних обрано реляційну систему керування базами даних MySQL, яка є стабільною, продуктивною та широко застосовується у веб-застосунках. Вибір реляційної моделі даних зумовлений чіткою структурою предметної області та наявністю логічних зв'язків між основними об'єктами системи навчального призначення. Такий підхід дозволяє ефективно реалізувати ієрархічну організацію навчального контенту та підтримувати узгодженість даних.

Взаємодія серверної частини системи з базою даних реалізується з використанням об'єктно-реляційного відображення Eloquent ORM, яке є складовою фреймворку Laravel. ORM надає можливість працювати з таблицями бази даних у вигляді програмних об'єктів, що значно спрощує реалізацію бізнес-логіки та підвищує читабельність коду. Замість безпосереднього формування SQL-запитів розробник оперує методами моделей, що зменшує ризик помилок та підвищує безпеку роботи з даними.

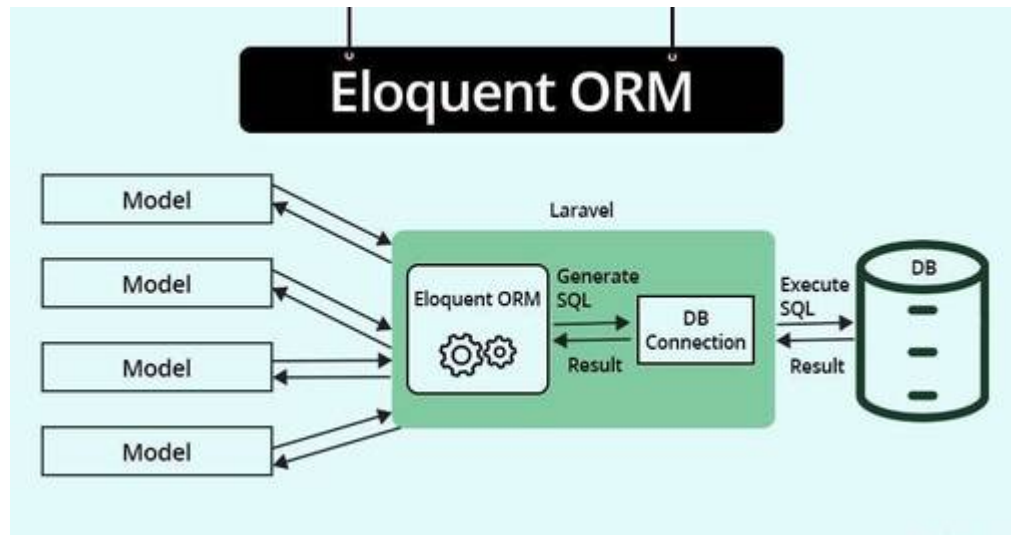


Рис. 3.5 Схематична ілюстрацію Laravel Eloquent

Кожна сутність предметної області системи відповідає окремій моделі. Так, для збереження даних про користувачів використовується модель користувача, яка містить атрибути для ідентифікації, автентифікації та збереження службової інформації. Навчальні курси реалізуються окремою моделлю, що дозволяє групувати навчальний контент за дисциплінами або напрямками підготовки. Теми та навчальні матеріали пов'язані з курсами і реалізують багаторівневу ієрархію, що забезпечує логічну структуру навчального процесу.

Окремі моделі призначені для роботи з тестами та результатами тестування. Такий поділ дозволяє зберігати структуру тестових завдань, контролювати зв'язок між тестами і відповідними темами, а також фіксувати результати проходження тестів кожним користувачем. Це створює передумови для подальшого аналізу навчального прогресу та оцінювання рівня засвоєння матеріалу.

Для забезпечення коректної роботи з базою даних у проєкті використовується механізм міграцій Laravel, який дозволяє описувати структуру таблиць у вигляді програмного коду. Міграції виконують роль інструменту керування версіями бази даних і дозволяють відтворювати її структуру у будь-якому середовищі розгортання. Кожна міграція визначає набір полів таблиці, типи даних, первинні ключі та обмеження цілісності.



У процесі проєктування бази даних особлива увага приділяється визначенню зв'язків між таблицями. Для цього використовуються зовнішні ключі, які забезпечують відповідність між записами у різних таблицях. Наприклад, кожна тема пов'язується з певним навчальним курсом, а кожен навчальний матеріал - з конкретною темою. Результати тестування пов'язуються як з користувачем, так і з тестом, що дозволяє однозначно визначити, хто і який тест проходив. Такий підхід гарантує логічну узгодженість даних та запобігає виникненню «висячих» записів.

Для підвищення ефективності доступу до даних у структурі бази даних передбачено використання індексів для полів, які найчастіше використовуються у пошукових запитах і при формуванні списків курсів, тем і матеріалів. Це дозволяє зменшити час виконання запитів і підвищити загальну продуктивність системи, особливо при збільшенні обсягу навчальної інформації.

Окрім основних навчальних даних, база даних системи містить службові таблиці, необхідні для реалізації механізмів автентифікації, керування сесіями та контролю доступу. Використання стандартних засобів Laravel у поєднанні з власними моделями дозволяє інтегрувати ці компоненти у загальну структуру без порушення логіки предметної області.

Завдяки застосуванню ORM та міграцій реалізація взаємодії з базою даних є гнучкою та придатною до розширення. У разі потреби система може бути доповнена новими сутностями або зв'язками без суттєвої перебудови існуючої структури. Це є важливою перевагою для систем навчального призначення, які повинні адаптуватися до змін освітніх вимог та розширення функціональних можливостей.

Отже, реалізація взаємодії з базою даних у системі навчального призначення базується на використанні реляційної моделі даних, ORM Eloquent та механізму міграцій Laravel, що забезпечує надійність зберігання

навчальної інформації, цілісність даних та зручність подальшого розвитку програмного продукту.

### **3.4. Реалізація основних модулів системи навчального призначення**

Програмна реалізація системи навчального призначення базується на модульному підході, який передбачає поділ функціональності програмного продукту на окремі логічно завершені компоненти. Такий підхід дозволяє спростити проєктування, реалізацію та подальший супровід системи, а також забезпечує можливість її розширення без суттєвої перебудови існуючої архітектури.

Кожен модуль системи відповідає за виконання окремих завдань навчального процесу та взаємодіє з іншими модулями через чітко визначені інтерфейси. Це забезпечує узгодженість роботи системи, зменшує залежність між компонентами та підвищує загальну стабільність програмного продукту. Реалізовані модулі відображають основні етапи організації навчальної діяльності - від ідентифікації користувачів до подання навчального контенту та обліку результатів навчання.

У межах даної роботи було реалізовано такі основні модулі системи навчального призначення:

1. Модуль авторизації та розмежування ролей користувачів Забезпечує вхід користувачів у систему та контроль доступу до функціональних можливостей відповідно до ролей «вчитель» і «студент».
2. Модуль управління користувачами Реалізує збереження даних користувачів, облік їх навчальної діяльності та взаємодію з іншими компонентами системи.
3. Модуль управління навчальними курсами Відповідає за створення, редагування та відображення навчальних курсів, а також за формування загального переліку доступних дисциплін.

4. Модуль структуризації навчального контенту Забезпечує ієрархічну організацію навчальних матеріалів за принципом «курс - тема - урок».
5. Модуль управління уроками та видами навчальної діяльності Дозволяє створювати уроки різних типів, зокрема лекції та практичні завдання, а також задавати параметри їх виконання.
6. Модуль роботи з навчальними матеріалами Реалізує завантаження, збереження та доступ до файлів навчального призначення, прив'язаних до конкретних уроків і тем.
7. Модуль журналу оцінок і обліку результатів навчання Забезпечує фіксацію результатів навчальної діяльності студентів і формування інформації для аналізу успішності.
8. Модуль навігації та інтерфейсу користувача Відповідає за зручність взаємодії користувача із системою, послідовність інтерфейсу та доступність функціональних елементів.

Представлений перелік модулів відображає логіку побудови системи навчального призначення та забезпечує повне охоплення основних функцій, необхідних для організації навчального процесу. Кожен із зазначених модулів реалізовано як окремий функціональний компонент у межах єдиної архітектури веб-застосунку.

У наступних підрозділах третього розділу буде здійснено детальний розгляд реалізації окремих модулів, зокрема описано їхню логіку роботи, взаємодію з базою даних та особливості використання в освітньому процесі.

Такий підхід дозволяє поетапно проаналізувати програмну реалізацію системи та продемонструвати її практичну придатність.

### **3.5. Реалізація модуля користувачів**

Модуль користувачів передбачає авторизацію та розмежування ролей користувачів та є одним з базових компонентів системи навчального

призначення, оскільки він забезпечує ідентифікацію користувачів, контроль доступу до функціональних можливостей системи та захист навчальної інформації від несанкціонованого використання. Коректна реалізація даного модуля є необхідною умовою стабільної та безпечної роботи всієї системи.

У розробленій системі реалізовано механізм автентифікації користувачів на основі облікових записів, які містять унікальні ідентифікаційні дані. Процес входу до системи здійснюється через окрему сторінку авторизації, де користувач вводить електронну пошту та пароль. Після перевірки коректності введених даних система встановлює сесію користувача та надає доступ до відповідних функціональних модулів.

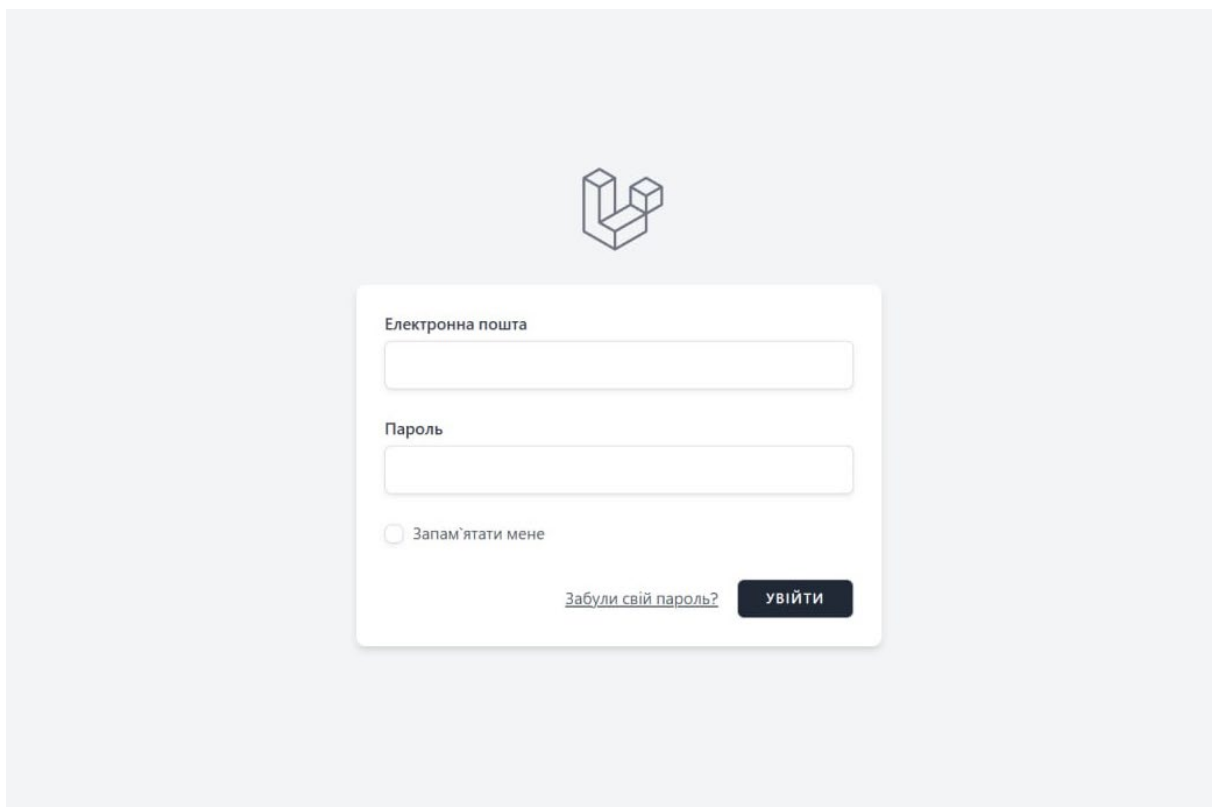


Рис. 3.6 Сторінка авторизації в системі

Особливістю реалізації є використання рольової моделі доступу. У системі визначено дві основні ролі користувачів: вчитель та студент. Кожна роль має власний набір дозволених дій, що дозволяє чітко розмежувати функціональність системи та уникнути некоректного використання ресурсів. Наприклад, лише користувач із роллю вчителя має можливість створювати та

редагувати навчальні курси, додавати теми й уроки, формувати практичні завдання та керувати навчальними матеріалами. Студент, у свою чергу, має доступ до перегляду курсів, навчальних матеріалів і виконання навчальних завдань.

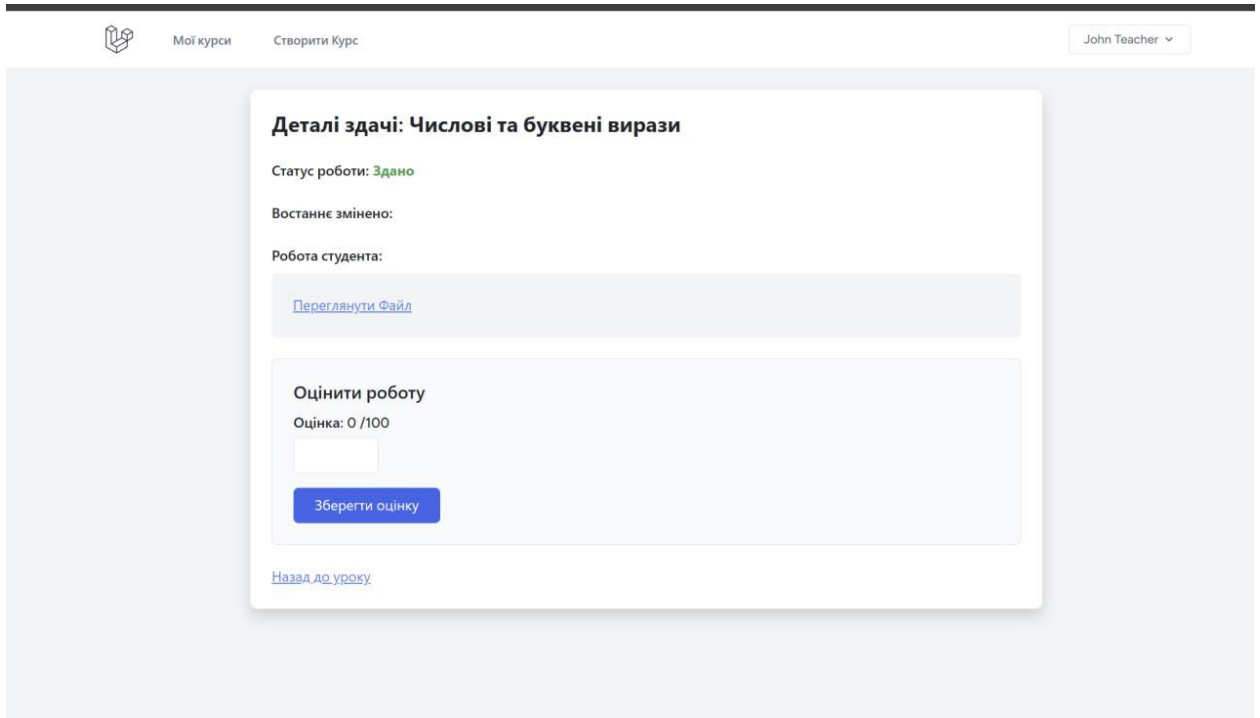


Рис. 3.7 Сторінка уроку якщо користувач авторизований як Вчитель

Реалізація рольової моделі доступу дозволяє динамічно керувати відображенням інтерфейсу користувача. Після входу до системи інтерфейс адаптується відповідно до ролі користувача: відображаються лише ті елементи керування та розділи меню, які доступні конкретній ролі. Такий підхід підвищує зручність користування системою та зменшує ризик помилкових дій з боку користувачів.

Початок приймання: 2025-12-13  
Термін спливає: 2025-12-25  
Файл з завданням: [Файл](#)

Статус роботи

Статус роботи	Не здано
Оцінка	—
Залишилося часу	4 Дні

Завантажити файл

Виберите файл

Файл не выбран

Здати роботу

Попередня діяльність

Натуральні, цілі, дробові числа

Перейти до...

Наступна діяльність

Відсоткові обчислення

Рис. 3.8 Сторінка уроку якщо користувач авторизований як учень.

На ілюстрації видно що користувач з роллю вчителя оглянути роботу учня, оцінити її чи повернути роботу на доопрацювання. А той час як користувач з роллю учня може лише переглянути завдання та здати його для перевірки та оцінки.

Модуль авторизації інтегровано з іншими компонентами системи, зокрема з модулем управління курсами, навчальними матеріалами та журналом оцінок. Це дозволяє забезпечити наскрізний контроль доступу до даних і функцій системи. Наприклад, під час звернення до сторінки редагування курсу система перевіряє роль користувача та обмежує доступ у разі відсутності відповідних прав.

Важливою складовою модуля є забезпечення захисту облікових даних користувачів. Паролі зберігаються у зашифрованому вигляді, що відповідає сучасним вимогам інформаційної безпеки. Крім того, реалізовано перевірку коректності введених даних та обробку помилкових спроб входу, що підвищує надійність роботи системи.

Також необхідно надати кожному користувачеві можливість змінювати деякі особисті дані, наприклад електронну адресу чи пароль. Для цього створена сторінка налаштування акаунту. На цій сторінці користувач може змінити лише свій E-mail, в той час ім'я змінити не можна, ця можливість є лише в користувача з роллю вчителя, як і можливість видалення акаунту.

Також для забезпечення безпеки та конфіденційності кожен користувач може змінити свій пароль в будь-який час, для цього потрібно скористатися блоком оновлення паролю, ввівши старий пароль один раз і новий пароль два рази.

Мої курси Jane Student ▾

### Профіль

#### Інформація про профіль

Оновіть інформацію про профіль та адресу електронної пошти вашого облікового запису.

Ім'я  
Jane Student

Електронна пошта  
student@email.com

**ЗБЕРЕГТИ**

#### Оновити пароль

Для забезпечення безпеки використовуйте довгий випадковий пароль для свого облікового запису.

Поточний пароль

Новий пароль

Пітвeрдіть пароль

**ЗБЕРЕГТИ**

#### Видалити Акаунт

Після видалення вашого облікового запису всі його ресурси та дані будуть остаточно видалені. Перед видаленням облікового запису завантажте всі дані та інформацію, які ви бажаєте зберегти.

**ВИДАЛИТИ АКАУНТ**

Рис.

### 3.9 Сторінка налаштування акаунту користувача

Загалом, реалізований модуль авторизації та ролей користувачів забезпечує надійну ідентифікацію учасників навчального процесу, чітко

розмежування функціональних можливостей та безпечний доступ до навчальної інформації. Його інтеграція з іншими модулями системи створює цілісне середовище для організації навчальної діяльності та слугує основою для реалізації більш складних функціональних компонентів, які розглядаються у наступних підрозділах.

### **3.6. Реалізація модуля управління курсами та навчальним контентом**

Модуль управління курсами та навчальним контентом є одним із ключових компонентів системи навчального призначення, оскільки саме він забезпечує формування навчального середовища, організацію матеріалів і доступ користувачів до освітніх ресурсів. Реалізація цього модуля спрямована на створення зручного та структурованого простору для подання навчального контенту відповідно до логіки навчального процесу.

У розробленій системі навчальні курси виступають основною одиницею організації контенту. Кожен курс представляє окрему навчальну дисципліну та містить загальну інформацію, таку як назва, опис і візуальне оформлення. Після авторизації користувач отримує доступ до сторінки зі списком курсів, які відображаються у вигляді карток. Такий підхід забезпечує наочність і спрощує навігацію в межах системи.

Для користувачів з роллю вчителя реалізовано можливість створення нових курсів і управління їх наповненням. Процес створення курсу передбачає задання основних параметрів дисципліни, після чого курс стає доступним у загальному переліку. Студенти, у свою чергу, мають можливість переглядати курси та ознайомлюватися з їхнім вмістом без доступу до функцій



редагування, що відповідає принципам розмежування ролей у системі.

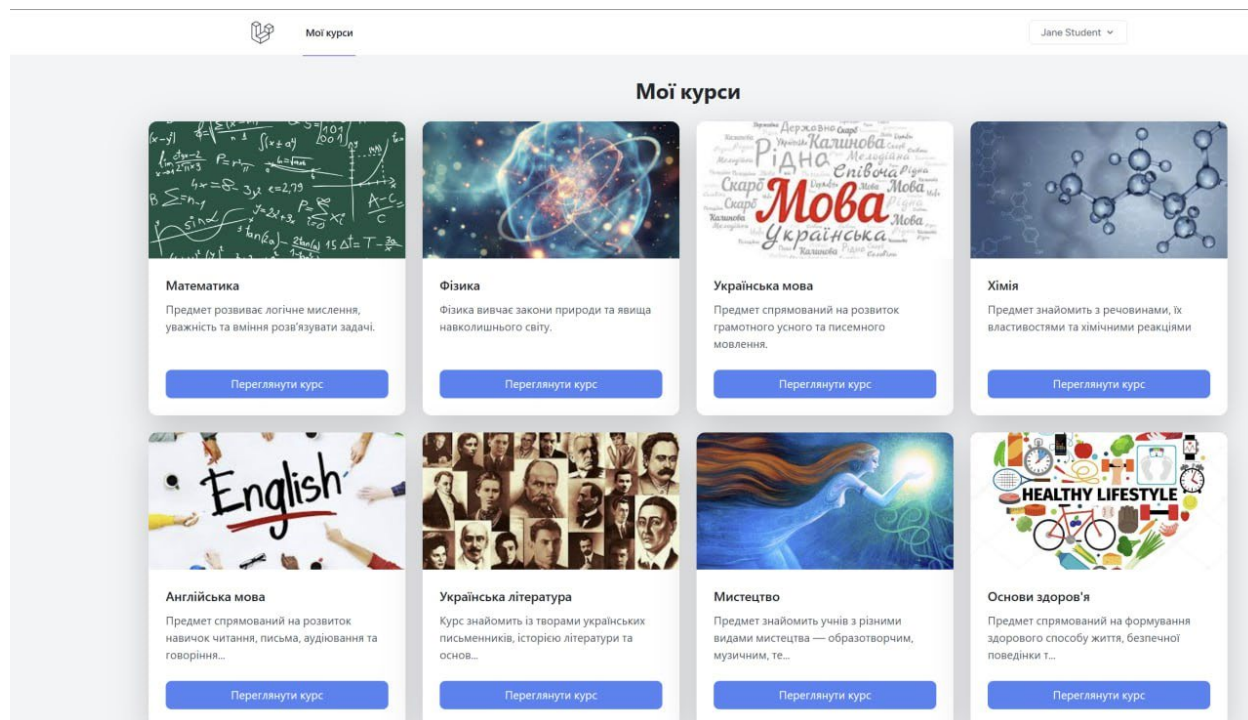


Рис. 3.11 Сторінка курсів.

Навчальний контент у межах курсу організовано за ієрархічною структурою, що включає теми та уроки. Темі використовуються для логічного поділу навчального матеріалу на змістові розділи, які відображають основні блоки навчальної програми. Кожна тема містить перелік уроків, які реалізують конкретні види навчальної діяльності. Така структура забезпечує послідовність подання матеріалу та відповідає традиційній організації навчального процесу.

Модуль управління навчальним контентом надає вчителю інструменти для створення, редагування та впорядкування тем і уроків. Зокрема, реалізовано можливість визначення порядку розміщення тем і уроків у межах курсу, що дозволяє адаптувати структуру контенту відповідно до методичних вимог або особливостей навчальної програми. Зміни в структурі курсу відображаються в інтерфейсі системи без потреби додаткових дій з боку користувача.

Навчальні матеріали прив'язуються безпосередньо до уроків, що дозволяє чітко пов'язати теоретичний і практичний контент з відповідними елементами курсу. Матеріали можуть бути представлені у вигляді файлів, що завантажуються в систему під час створення або редагування уроку. Такий підхід забезпечує централізоване зберігання навчальних ресурсів і спрощує доступ до них для студентів.

Важливим аспектом реалізації модуля є адаптація інтерфейсу користувача залежно від ролі. Для вчителя інтерфейс містить елементи керування, що дозволяють додавати нові теми й уроки, тоді як для студента відображається лише навчальний контент у режимі перегляду. Це сприяє зручності використання системи та запобігає виконанню небажаних дій.

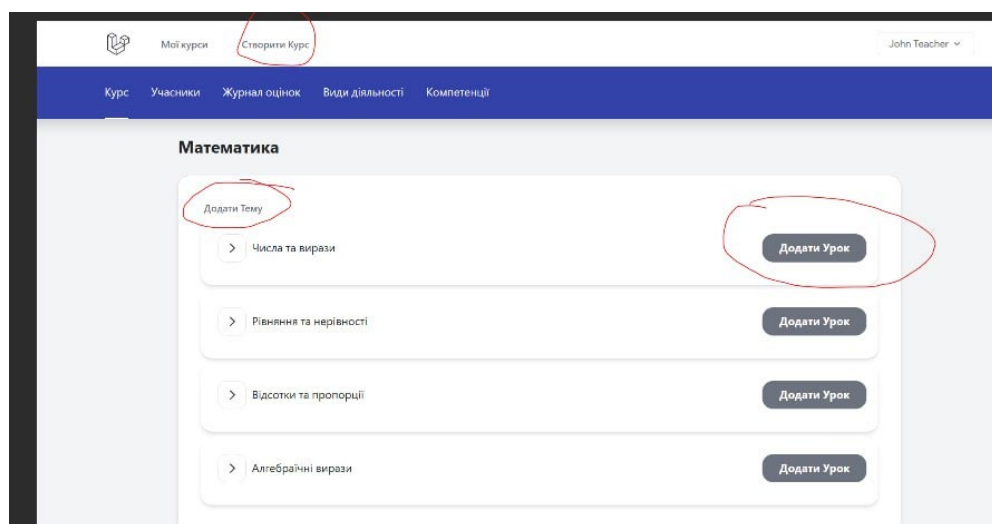


Рис. 3.12 Сторінка курсу математика для ролі Вчитель

Усі елементи навчального контенту інтегровані з базою даних системи, що забезпечує збереження структури курсів і матеріалів між сесіями роботи та дозволяє відновлювати навчальне середовище після повторного входу до системи. Взаємодія модуля управління курсами з іншими компонентами, зокрема з модулем авторизації та журналом оцінок, забезпечує цілісність і узгодженість функціонування системи.

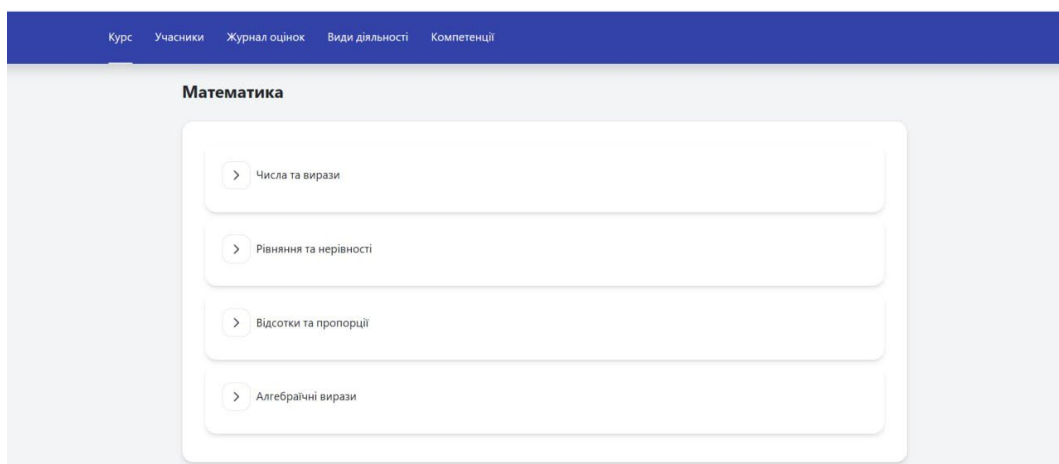


Рис. 3.13 Сторінка курсу математика для ролі Учень

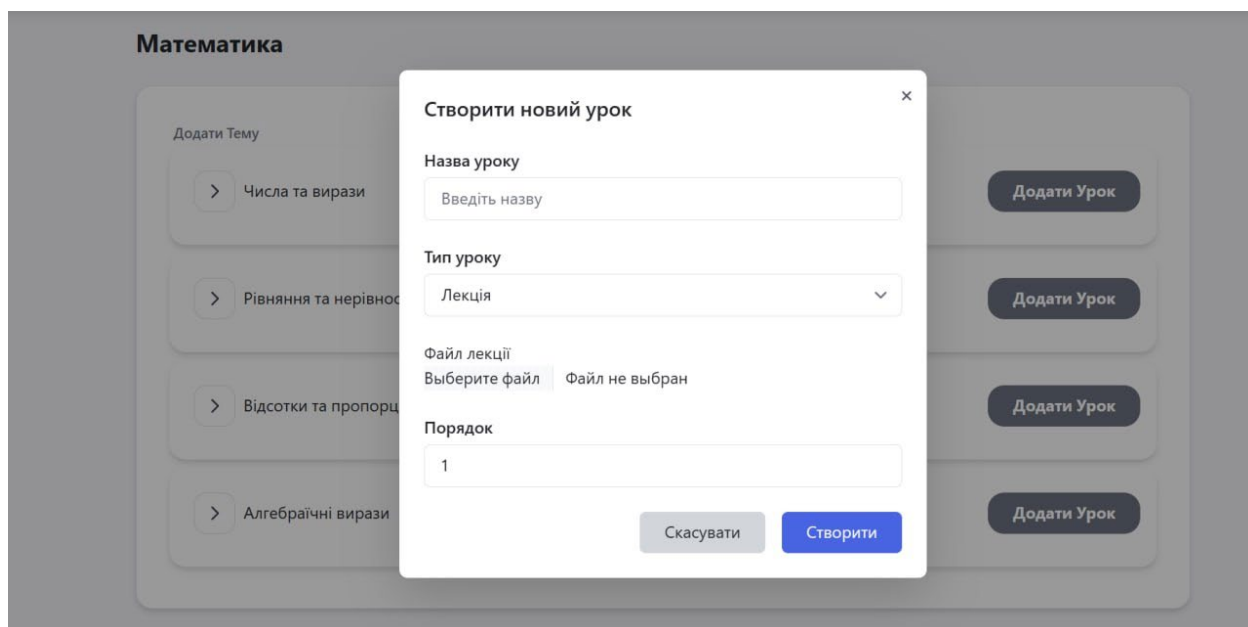
Таким чином, реалізований модуль управління курсами та навчальним контентом забезпечує повноцінну організацію навчального матеріалу, підтримує логічну структуру курсів і створює зручні умови для взаємодії вчителів і студентів у межах системи навчального призначення.

### 3.7. Реалізація модуля уроків і видів навчальної діяльності

Модуль уроків і видів навчальної діяльності реалізує ключову частину функціональності системи навчального призначення, оскільки саме на цьому рівні відбувається безпосередня взаємодія користувачів із навчальним контентом. Даний модуль забезпечує створення, налаштування та організацію окремих уроків у межах тем і курсів, а також підтримує різні форми навчальної діяльності відповідно до педагогічних завдань.

У розробленій системі урок є логічно завершеною одиницею навчального процесу, яка входить до складу певної теми курсу. Такий підхід дозволяє забезпечити чітку ієрархію навчального контенту та сприяє послідовному засвоєнню матеріалу. Для користувача з роллю вчителя реалізовано зручний механізм створення уроків безпосередньо з інтерфейсу курсу, що зменшує кількість дій, необхідних для наповнення навчального середовища.

Процес створення нового уроку здійснюється за допомогою спеціальної форми, яка відкривається у вигляді модального вікна. Така реалізація дозволяє вчителю працювати з навчальним контентом, не залишаючи сторінку курсу, що підвищує зручність користування системою. Під час створення уроку вчитель задає назву уроку, обирає тип навчальної діяльності та визначає порядковий номер уроку в межах теми, що забезпечує логічну послідовність подання матеріалу.



The image shows a web interface for creating a new lesson. In the background, there is a sidebar titled "Математика" (Mathematics) with a "Додати Тему" (Add Topic) button and a list of topics: "Числа та вирази" (Numbers and expressions), "Рівняння та нерівності" (Equations and inequalities), "Відсотки та пропорції" (Percentages and proportions), and "Алгебраїчні вирази" (Algebraic expressions). The main area displays a modal window titled "Створити новий урок" (Create new lesson). This modal contains the following fields: "Назва уроку" (Lesson name) with a text input field containing the placeholder "Введіть назву" (Enter name); "Тип уроку" (Lesson type) with a dropdown menu currently showing "Лекція" (Lecture); "Файл лекції" (Lesson file) with a "Виберіть файл" (Choose file) button and a status "Файл не вибран" (File not selected); and "Порядок" (Order) with a text input field containing the number "1". At the bottom of the modal are two buttons: "Скасувати" (Cancel) and "Створити" (Create). To the right of the modal, there are four "Додати Урок" (Add Lesson) buttons stacked vertically.

Рис. 3.14 форма для створення уроку.

Важливою особливістю модуля є підтримка різних типів уроків, що відповідають різним видам навчальної діяльності. Зокрема, система підтримує уроки типу лекція та практичне завдання. Лекційні уроки використовуються для подання теоретичного матеріалу, тоді як практичні завдання спрямовані на закріплення знань і формування практичних навичок. Такий поділ дозволяє реалізувати змішану модель навчання та адаптувати курс до особливостей конкретної дисципліни.

Для практичних завдань у формі створення уроку передбачено додаткові поля, які дозволяють вчителю завантажити файл із завданням, додати текстовий опис та встановити часові межі приймання робіт. Наявність полів

для задання дати початку та завершення приймання дозволяє організувати навчальну діяльність у часі та дисциплінує студентів щодо виконання завдань у встановлені строки. Це особливо важливо для реалізації контролю навчальної діяльності в дистанційному форматі.

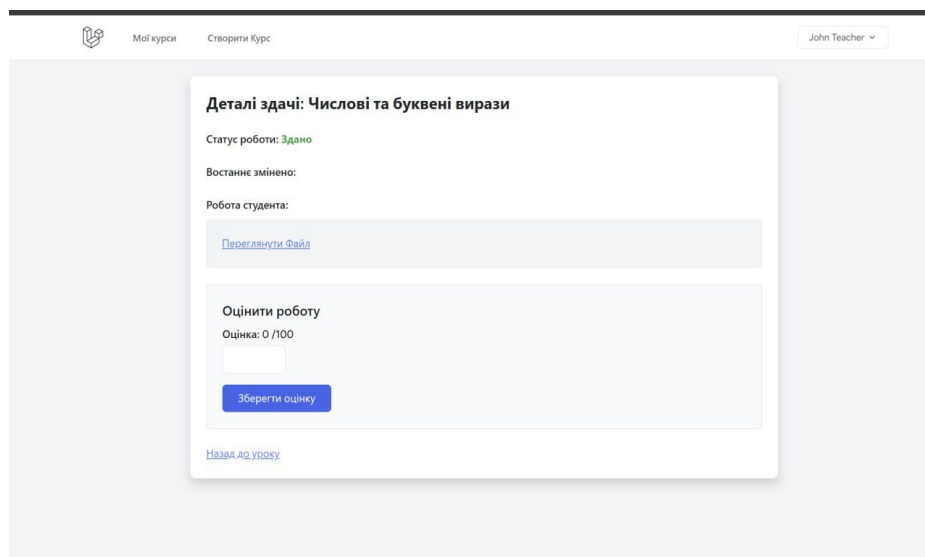


Рис. 3.15 Сторінка детального огляду роботи студента

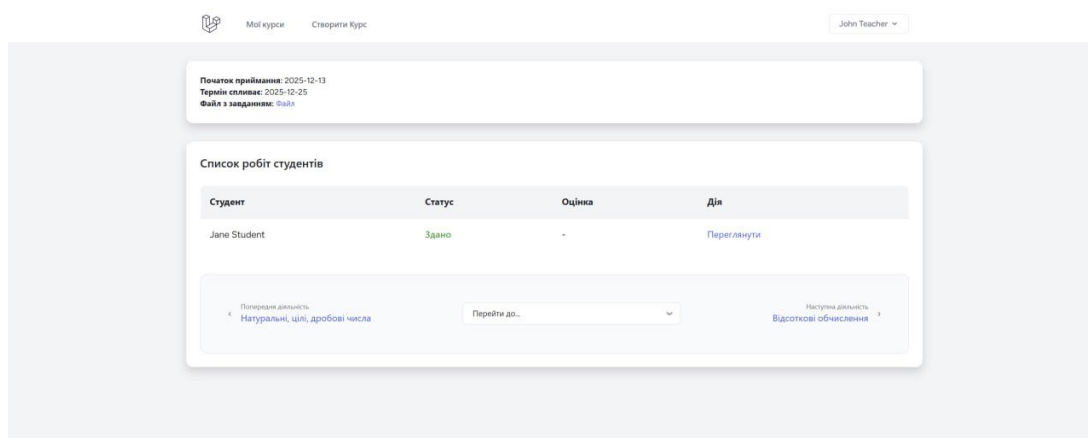


Рис. 3.16 Сторінка зі списком робіт учнів з уроку

Модуль уроків тісно інтегрований з модулем управління курсами та темами. Кожен урок створюється в контексті конкретної теми, що відображається в інтерфейсі курсу у вигляді структурованого списку. Для вчителя передбачено окремі елементи керування, такі як кнопки додавання тем і уроків, що дозволяють оперативно змінювати структуру курсу. Студенти,

у свою чергу, бачать лише впорядкований перелік доступних уроків без можливості редагування.

Інтерфейс модуля уроків розроблено з урахуванням принципів наочності та мінімалізму. Всі елементи керування мають зрозумілі підписи та логічне розташування, що зменшує когнітивне навантаження на користувачів і спрощує процес наповнення курсу. Використання однакових форм і підходів для різних типів уроків забезпечує уніфікованість інтерфейсу та сприяє швидкому освоєнню системи.

Початок приймання: 2025-12-13  
Термін спливає: 2025-12-25  
Файл з завданням: [Файл](#)

### Статус роботи

Статус роботи	Не здано
Оцінка	—
Залишилося часу	4 Дні

Завантажити файл

Файл не выбран

← Попередня діяльність  
Натуральні, цілі, дробові числа

Перейти до...

Наступна діяльність  
Відсоткові обчислення →

Рис. 3.17 Сторінка для здачі роботи для учня

Таким чином, реалізований модуль уроків і видів навчальної діяльності забезпечує гнучке та ефективне управління навчальним процесом на рівні окремих занять. Підтримка різних типів уроків, можливість задання параметрів виконання та тісна інтеграція з іншими модулями системи дозволяють створити повноцінне навчальне середовище, придатне для використання в умовах дистанційного та змішаного навчання. Отримані рішення створюють основу для подальшої реалізації механізмів оцінювання та

обліку результатів навчальної діяльності, що буде розглянуто у наступному підрозділі.

### **3.8. Реалізація модуля журналу оцінок і результатів навчання**

Модуль журналу оцінок і результатів навчання є завершальним елементом функціонального циклу системи навчального призначення, оскільки він забезпечує облік навчальної діяльності студентів, фіксацію результатів виконання завдань та формування узагальненої інформації про успішність. Реалізація даного модуля спрямована на підтримку процесів контролю знань і прозорості взаємодії між учасниками навчального процесу.

У розробленій системі журнал оцінок тісно інтегрований з модулями уроків і видів навчальної діяльності. Кожне практичне завдання, створене в межах курсу, автоматично формує відповідний запис у журналі, що дозволяє пов'язати результати навчальної діяльності з конкретним уроком, темою та курсом. Такий підхід забезпечує цілісність даних і спрощує аналіз успішності студентів.

Доступ до журналу оцінок реалізовано з урахуванням ролей користувачів. Користувач із роллю вчителя має можливість переглядати результати виконання завдань усіма студентами, аналізувати їхню активність та фіксувати оцінки. Для студента журнал виконує інформаційну функцію - він дозволяє переглядати власні результати навчання та відстежувати виконання практичних завдань у межах курсу.

Журнал оцінок до завдання: Пропорції		
Учень	Оцінка	Буквенна оцінка
Jane Student	90	A
Wiley Student	65	C
Wiley Student	71	C
John Student	43	F

Рис. 3.18 Відображення оцінок курсу для вчителя.

Модуль журналу оцінок забезпечує збереження інформації про результати навчальної діяльності у базі даних системи. Для кожного запису фіксується зв'язок із користувачем, відповідним уроком і видом діяльності, що дозволяє формувати детальні вибірки та узагальнені звіти. Така структура даних створює умови для подальшого розширення функціоналу, зокрема додавання підсумкових оцінок або аналітики навчального прогресу.

Важливим аспектом реалізації модуля є наочність подання інформації. Дані журналу оцінок відображаються у структурованому вигляді, що дозволяє швидко оцінити рівень виконання завдань і загальну успішність. Чітка прив'язка оцінок до конкретних уроків і тем полегшує орієнтацію в навчальних результатах і сприяє прозорості оцінювання.


 Мої курси		Jane Student ▾
Журнал оцінок		
Домашнє завдання	Оцінка	Буквенна оцінка
Числові та буквені вирази	90	A
Рівняння з однією змінною	72	C
Пропорції	48	F

Рис. 3.19 Відображення журналу для учня



Модуль журналу оцінок також виконує роль зворотного зв'язку в навчальному процесі. Наявність зафіксованих результатів дозволяє студентам оцінювати власний прогрес і коригувати навчальну діяльність, а вчителям - аналізувати ефективність поданого матеріалу та складність завдань. Це підвищує педагогічну цінність системи та сприяє більш усвідомленому навчанню.

Інтеграція журналу оцінок з іншими модулями системи забезпечує узгодженість функціонування програмного продукту в цілому. Результати навчальної діяльності зберігаються між сеансами роботи, доступні для перегляду відповідним користувачам і можуть використовуватися як основа для формування підсумкових висновків щодо успішності навчання.

Таким чином, реалізований модуль журналу оцінок і результатів навчання забезпечує систематичний облік навчальної діяльності, підтримує процеси контролю та аналізу знань і завершує функціональну модель системи навчального призначення. Його використання дозволяє створити цілісне навчальне середовище, орієнтоване на досягнення освітніх результатів та підвищення ефективності навчального процесу.

### **3.9 Структури бази даних для навчальних модулів**

Ефективність роботи системи навчального призначення значною мірою визначається коректно спроектованою структурою бази даних, яка забезпечує збереження навчального контенту, облік користувачів, фіксацію їхньої навчальної активності та формування результатів навчання. Для реалізації зазначених завдань у проєкті використано реляційну модель даних, що є доцільною через чіткі зв'язки між основними сутностями: користувачами, курсами, розділами (темами), уроками, роботами студентів та оцінюванням.

ER-діаграма відображає структуру даних в БД подібну до підходів, що застосовуються сучасних LMS-системах (наприклад, Moodle або Google

Classroom), і охоплює повний цикл взаємодії учасників навчання: від створення курсу та наповнення контентом до подання робіт студентами та виставлення оцінок (див. Додаток А).

#### *Таблиця users (користувачі)*

Таблиця users є базовою сутністю системи та зберігає інформацію про всіх користувачів незалежно від їхньої ролі. У структурі таблиці передбачено атрибути для ідентифікації (id), автентифікації (E-mail, password) та відображення даних користувача (name). Роль користувача задається полем role, яке приймає значення teacher або student. Це дозволяє реалізувати рольову модель доступу та визначити права користувача на виконання операцій у системі.

Наявність полів created\_at та updated\_at забезпечує ведення історії створення й оновлення записів, що є корисним для контролю змін та аудиту даних.

#### *Таблиця courses (курси)*

Таблиця courses описує навчальні курси, які створюються та адмініструються користувачами з роллю вчителя. Основні атрибути включають ідентифікатор курсу (id), назву (title) та опис (description). Зв'язок між курсом і вчителем реалізовано через поле teacher\_id, яке є зовнішнім ключем і посилається на users.id. Таким чином забезпечується правило: кожен курс має одного власника-вчителя, а один учитель може створювати декілька курсів.

Додатково в таблиці передбачено поле code, яке може використовуватися як код доступу для приєднання студентів до курсу. Це спрощує процедуру запису на курс та забезпечує керованість доступу.

### *Таблиця sections (розділи/теми курсу)*

Для структурування навчального контенту всередині курсу використано таблицю sections. Вона відображає поділ курсу на теми, модулі або тижні навчання. Поле course\_id формує зв'язок «один-до-багатьох» між курсом і його розділами: один курс містить багато розділів, а кожен розділ належить лише одному курсу.

Поле order\_num забезпечує можливість впорядкування розділів у заданій послідовності. Це важливо для підтримки логіки викладу матеріалу (наприклад, «Вступ» → «Тема 1» → «Тема 2» тощо). Наявність title та description дозволяє робити структуру курсу зрозумілою для користувачів.

### *Таблиця lessons (уроки/матеріали)*

Таблиця lessons реалізує зберігання окремих елементів навчального контенту в межах розділів. Кожен урок належить певному розділу через поле section\_id, що формує зв'язок «один-до-багатьох» між sections і lessons.

Зміст уроку може зберігатися в полі content, що дозволяє підтримувати текстові матеріали або HTML-контент. Тип уроку визначається полем lesson\_type, яке містить значення video, text, quiz, assignment. Така класифікація дозволяє системі відображати уроки різними способами та застосовувати до них різні сценарії взаємодії (перегляд матеріалу, проходження тесту, виконання завдання тощо). Поле order\_num забезпечує контроль порядку уроків у межах розділу.

Таким чином, зв'язка courses → sections → lessons формує ієрархічну модель, зручну для навігації та масштабування.

### *Таблиця course\_user (запис на курс)*

Для реалізації участі студентів у курсах застосовано проміжну таблицю course\_user, яка реалізує зв'язок типу «багато-до-багатьох» між курсами та

користувачами. У таблиці зберігаються пари `course_id` та `user_id`, а також час приєднання `enrolled_at`.

Унікальний індекс (`course_id`, `user_id`) гарантує, що один і той самий студент не може бути зарахований на один курс більше одного разу. Це забезпечує коректність даних і спрощує облік учасників курсу.

#### *Таблиця submissions (роботи студентів)*

Для підтримки практичної діяльності та контролю знань передбачено таблицю `submissions`, яка зберігає відповіді або виконані роботи студентів. Запис у таблиці пов'язується з конкретним уроком через `lesson_id`. Це дозволяє використовувати механізм подання робіт як для типу `assignment`, так і для `quiz` (якщо тестування реалізується через подання відповідей).

Поле `user_id` фіксує автора роботи (студента), а `content` містить текст відповіді або інші дані подання. Поле `submitted_at` дозволяє контролювати час виконання та застосовувати дедлайни в логіці системи.

#### *Таблиця grades (оцінювання)*

Таблиця `grades` реалізує механізм оцінювання поданих робіт та формування зворотного зв'язку. Ключовим є зв'язок із таблицею `submissions` через `submission_id`, який гарантує, що оцінка належить конкретному поданню студента.

Поле `grade` містить значення оцінки (числовий бал), а поле `feedback` призначене для текстового коментаря вчителя. Поле `graded_by` є зовнішнім ключем на `users.id` і фіксує вчителя, який здійснив оцінювання. Поле `graded_at` відображає момент виставлення оцінки. Така структура дозволяє вести облік результатів навчання та забезпечувати прозорість оцінювання.

Розроблена ER-модель забезпечує цілісний цикл навчальної взаємодії:

- Вчитель створює курс (`courses.teacher_id → users.id`), формує структуру курсу через розділи (`sections`) та наповнює її уроками (`lessons`).
- Студенти приєднуються до курсів через таблицю запису (`course_user`).
- Для уроків типу практичного завдання або тесту студент створює подання (`submissions`), після чого вчитель оцінює подання (`grades`) і надає зворотний зв'язок.

Така структура бази даних є логічною, нормалізованою та достатньо гнучкою для подальшого розширення системи (наприклад, додавання файлів до подань, введення журналу успішності на рівні курсу, підтримка кількох викладачів у курсі або введення компетентностей).

### **3.10. Реалізація механізму завантаження та перенесення курсу на пристрій користувача у вигляді архіву**

Однією з практично важливих можливостей системи навчального призначення є підтримка перенесення навчального контенту між пристроями та створення резервних копій курсів. Для цього до системи було передбачено механізм експорту/імпорту курсу у вигляді архіву, який дозволяє користувачу (переважно вчителю) завантажити курс на власний пристрій, а в подальшому - відновити або перенести його в інше середовище без ручного повторного створення структури й матеріалів.

#### **3.10.1. Призначення механізму архівації**

Завантаження курсу у форматі архіву використовується для:

- створення резервної копії навчального курсу;
- перенесення курсу між різними інсталяціями системи;
- передавання курсу іншому викладачу/адміністратору;
- швидкого відновлення структури та матеріалів курсу після змін або помилок.

### 3.10.2. Логіка експорту курсу

Експорт курсу реалізується як формування архівного файлу (наприклад, .zip), який містить: 1. метадані курсу (назва, опис, налаштування); 2. структуру розділів і уроків (порядок, типи занять); 3. вкладені навчальні матеріали (файли лекцій, завдань, додатки); 4. за потреби - параметри завдань (строки виконання, опис, додаткові поля).

При формуванні архіву система вибірково отримує дані з бази даних (курси, секції, уроки та пов'язані записи) і створює структурований пакет, у якому дані зберігаються у вигляді одного або декількох файлів опису (наприклад, JSON) та папок з матеріалами.

### 3.10.3. Структура архіву

Для забезпечення читабельності й стандартизації доцільно використовувати таку структуру:

- course.json - основні метадані курсу (id, title, description, code тощо);
- sections.json - список тем/розділів із полями title, description, order\_num;
- lessons.json - список уроків із полями section\_ref, title, lesson\_type, order\_num, content;
- materials/ - каталог файлів навчальних матеріалів (лекції, вкладення);
- • assignments/ - файли практичних завдань (за наявності);
- manifest.json - службовий файл з версією формату експорту та контрольними сумами (опційно).

Такий підхід забезпечує незалежність контенту від внутрішніх ідентифікаторів конкретної інсталяції та спрощує імпорт у іншу систему.

### 3.10.4. Логіка імпорту курсу

Імпорт передбачає завантаження архіву користувачем (вчителем) через інтерфейс системи та виконання послідовних етапів: 1. Перевірка архіву (формат, наявність обов'язкових файлів, версія структури). 2. Розпакування у тимчасове сховище та первинна валідація даних. 3. Створення нового курсу або оновлення існуючого (залежно від сценарію імпорту). 4. Додавання розділів (sections) із відновленням порядку. 5. Додавання уроків (lessons) із прив'язкою до відповідних розділів. 6. Копіювання файлів матеріалів у файлове сховище системи та відновлення зв'язків у БД. 7. Підсумкова перевірка цілісності (чи всі уроки мають правильні матеріали, чи збережено порядок).

Після завершення імпорту курс стає доступним у системі у звичному вигляді та може використовуватися без додаткового ручного налаштування.

### **3.10.5. Забезпечення коректності та безпеки**

Оскільки механізм імпорту працює з файлами, важливо передбачити базові заходи безпеки:

- обмеження типів файлів, які дозволено імпортувати як матеріали;
- перевірку розміру архіву та загального обсягу розпакованих файлів;
- збереження файлів у службове сховище з унікальними іменами, а не з оригінальними назвами;
- контроль доступу: імпорт/експорт доступний лише ролі вчителя або адміністратора;
- валідацію вмісту \*.json (структура, обов'язкові поля), щоб запобігти некоректним даним.

### **3.10.6. Переваги для освітнього процесу**

Реалізований механізм експорту/імпорту курсу у вигляді архіву підвищує практичну цінність системи навчального призначення, оскільки:

- спрощує розповсюдження навчальних матеріалів між викладачами;
- дозволяє швидко масштабувати курси й повторно використовувати їх у наступних навчальних роках;
- забезпечує резервне копіювання без потреби прямого доступу до бази даних; підтримує збереження структури курсу разом із матеріалами в єдиному пакеті.



### **Висновки до розділу 3**

У третьому розділі магістерської роботи було здійснено практичну реалізацію та апробацію розробленої системи навчального призначення, орієнтованої на зберігання, подання та використання навчальних матеріалів в умовах дистанційного та автономного навчання. Реалізація системи ґрунтувалася на результатах теоретичного аналізу, поданого в попередніх розділах, та враховувала сучасні вимоги до цифрових освітніх середовищ.

У ході розроблення було спроектовано структуру системи, що забезпечує логічну організацію навчального контенту, можливість навігації між розділами, а також зручний доступ до навчальних матеріалів незалежно від стабільності інтернет-з'єднання. Особливу увагу приділено реалізації автономного режиму роботи, який дозволяє використовувати систему в умовах обмеженого або відсутнього доступу до мережі Інтернет.

Практичне тестування системи засвідчило її працездатність та відповідність поставленим вимогам. Система забезпечує базові функції навчального призначення: подання теоретичного матеріалу, структуроване збереження інформації, підтримку навчальної діяльності та можливість тривалого використання навчальних ресурсів на локальному комп'ютері користувача.

Результати апробації підтвердили, що розроблена система є доцільною для використання в освітньому процесі, зокрема в умовах дистанційного та змішаного навчання. Вона може бути застосована як допоміжний інструмент для організації самостійної роботи здобувачів освіти, а також як елемент цифрового освітнього середовища закладу освіти.

Таким чином, поставлені в третьому розділі завдання було виконано, а отримані результати підтверджують практичну значущість і можливість подальшого розвитку розробленої системи навчального призначення.

## ВИСНОВКИ

У ході виконання магістерської роботи повністю досягнуто поставленої мети - спроектовано та реалізовано систему навчального призначення, орієнтовану на потреби дистанційної освіти. Проведений комплексний аналіз наукових джерел, нормативних документів та сучасних практичних рішень у сфері електронного навчання сформував цілісне уявлення про сучасні цифрові освітні системи. З'ясовано роль і місце систем навчального призначення в освітньому процесі, показано їх багатофункціональність (підтримка навчальної діяльності, управління освітнім процесом, збереження матеріалів) та встановлено відсутність єдиної універсальної класифікації таких систем.

Проаналізовано еволюцію дистанційної освіти в Україні та світі, дистанційне навчання перетворилося на повноцінний елемент освітніх систем, інтегрований у національні стратегії цифровізації, хоча в Україні його розвиток стримується низкою викликів (неоднакова цифрова інфраструктура, різний рівень цифрових компетентностей педагогів, нестабільні умови).

Окрім того, проведено огляд популярних LMS-платформ (Moodle, Google Classroom, Microsoft Teams, Canvas, Blackboard) з виявленням їх переваг і недоліків. Встановлено, що більшість сучасних платформ орієнтовані на постійне інтернет-з'єднання, що знижує їх ефективність за кризових умов нестабільного зв'язку. Узагальнення результатів теоретичного і прикладного аналізу дозволило виокремити ключові проблеми розроблення освітніх систем: залежність від інтернету, обмежена автономність, недостатня інтерактивність, проблеми інформаційної безпеки та низький рівень цифрової грамотності користувачів.

Ці висновки підтвердили актуальність створення нової системи, яка б відповідала вимогам стійкості, гнучкості та адаптивності сучасної освіти в Україні.

На підставі проведеного дослідження сформульовано вимоги до програмного продукту і спроектовано його архітектуру та структуру бази

даних. Розроблена система поєднує класичну архітектуру LMS з орієнтацією на структуроване зберігання навчальних матеріалів та підтримку різних видів навчальної діяльності, а також забезпечує можливість локального збереження й перенесення навчальних курсів у вигляді архівів для автономного використання. У рамках роботи реалізовано основні функціональні модулі системи: керування користувачами, курсами, розділами та уроками, інтеграція практичних завдань та тестів, ведення журналу оцінок тощо. Створений програмний продукт успішно виконує покладені на нього функції, що підтверджено внутрішнім тестуванням його працездатності. Фактична реалізація програмного комплексу підтверджує досягнення всіх завдань дослідження, визначених у вступі: виконано аналіз літератури, визначено тенденції й проблеми, сформовано вимоги, спроектовано систему та її базу даних, розроблено програмне забезпечення і перевірено його роботу. Наукова новизна розробки полягає в запропонованому підході до побудови системи електронного навчання, що забезпечує автономність і гнучкість використання: на відміну від відомих аналогів, система може ефективно функціонувати за нестабільного Інтернету та адаптуватися до різних освітніх потреб.

В цілому, досягнуті результати повністю відповідають поставленій меті і завданням магістерської роботи, а створена система навчального призначення підтвердила свою ефективність та перспективність для застосування у сфері дистанційної освіти.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Вакалюк Т. А. Види та призначення електронних засобів навчання // *Автоматизація та комп'ютерно-інтегровані технології у виробництві та освіті: стан, досягнення, перспективи розвитку*. 2013. С. 110-112.
2. Шишкіна М. П. Класифікація програмних засобів навчального призначення. Електронна бібліотека НАПН України. URL: <https://lib.iitta.gov.ua/352/1/Classif-Kir.pdf> (дата звернення: 17.09.2025).
3. Галчанська В. В. Значення електронних засобів навчального призначення та перспективи їх використання // *Українські студії в європейському контексті*. 2023. № 7. С. 201-206.
4. Гаврілова Л. М., Катасонова Ю. О. (2017). Теоретичні аспекти впровадження дистанційного навчання в Україні. *Освітологічний дискурс*. № 1-2 (16-17). С. 168-177. URL: <https://od.kubg.edu.ua/index.php/journal/article/view/433> (дата звернення: 17.09.2025).
5. Биков В. Ю. Дистанційна освіта: актуальність, принципи, технології. *Інформаційні технології і засоби навчання*. 2008 № 2. URL: <https://lib.iitta.gov.ua/707> (дата звернення: 17.09.2025).
6. Кухаренко В. М. (2015). Системний підхід до змішаного навчання. *Інформаційні технології в освіті*. № 24. С. 53-67. URL: <https://ite.kspu.edu/index.php/ite/article/view/568> (дата звернення: 17.09.2025).
7. Литвинова С. Г. (2017). Сучасні електронні освітні середовища: стан і перспективи. *Інформаційні технології і засоби навчання*. № 5. URL: <https://lib.iitta.gov.ua/708> (дата звернення: 27.09.2025).
8. Спірін О. М. (2020). Тенденції розвитку цифрової компетентності педагогів. *Інформаційні технології і засоби навчання*. № 2. URL: <https://lib.iitta.gov.ua/718> (дата звернення: 27.09.2025).
9. Moore M. G. (Ed.) (2013). *Handbook of Distance Education* (3rd ed.). New York: Routledge.

10. Anderson T. (2016). The Theory and Practice of Online Learning. Athabasca University Press. URL: <https://www.aupress.ca/books/120146-theory-and-practice-of-online-learning/> (дата звернення: 27.09.2025)
11. Means B., Toyama Y., Murphy R., Bakia M., Jones K. (2010). Evaluation of Evidence-Based Practices in Online Learning. U.S. Department of Education. URL: <https://www2.ed.gov/rschstat/eval/tech/evidence-based-practices/finalreport.pdf> (дата звернення: 29.09.2025).
12. Garrison D. R. E-Learning in the 21st Century. London: Routledge.
13. UNESCO (2020). *Education in a post-COVID world*. URL: <https://unesdoc.unesco.org/ark:/48223/pf0000373717> (дата звернення: 03.10.2025)
14. OECD (2021). *The State of School Education: One Year into the COVID Pandemic*. URL: <https://www.oecd.org/education/state-school-education-covid/> (дата звернення: 03.10.2025)
15. EDUCAUSE (2019). *Horizon Report: Higher Education Edition*. URL: <https://library.educause.edu/resources/2019/4/2019-horizon-report> (дата звернення: 03.10.2025)
16. Anderson T. The Theory and Practice of Online Learning. Athabasca University Press, 2016. P. 41-44.
17. Garrison D. R. E-Learning in the 21st Century. London: Routledge, 2017. P. 28-31.
18. Reich J. Rebooting MOOC Research. Science. 2015. Vol. 347(6217). P. 23-25
19. UNESCO. Education in a Post-COVID World. Paris, 2020. P. 6-9.
20. OECD. The State of School Education: One Year into the COVID Pandemic. Paris, 2021. P. 33-36.
21. Systematic Comparative Review of Three LMS Platforms: Moodle, Blackboard and Canvas. 2024. P. 3-12.
22. Comparative Study between Moodle, Canvas and Google Classroom. 2025. P. 41-50.
23. Nafidi Y., El Batri B. A Comparative Analysis of Moodle and Google Classroom. 2023. P. 112-118.

24. Rogers J. K. B. Learning Management Systems in Higher Education. ERIC, 2025.
25. Moodle and Google Classroom: a comparative study of acceptability. 2025. P. 18-23.
26. Яненко Я. Застосування Microsoft Teams у дистанційному навчанні. Інформаційні технології і засоби навчання, 2024, Том 100, №2
27. Кремень, В. Г., Биков, В. Ю., Ляшенко, О. І., Литвинова, С. Г., Луговий, В. І., Мальований, Ю. І., ... Топузов, О. М. (2022). НАУКОВО-МЕТОДИЧНЕ ЗАБЕЗПЕЧЕННЯ ЦИФРОВІЗАЦІЇ ОСВІТИ УКРАЇНИ: СТАН, ПРОБЛЕМИ, ПЕРСПЕКТИВИ: Наукова доповідь загальним зборам НАПН України «Науково-методичне забезпечення цифровізації освіти України: стан, проблеми, перспективи», 18-19 листопада 2022 р. Вісник Національної академії педагогічних наук України, 4(2), 1-49. <https://doi.org/10.37472/v.naes.2022.4223>.
28. OECD. *Digital Education Outlook 2021: Pushing the Frontiers with Artificial Intelligence, Blockchain and Robots*. Paris : OECD Publishing, 2021. P. 22-34.
29. European Commission. *Digital Education Action Plan 2021-2027*. Brussels, 2020. P. 15-18.
30. Аксак Н. Г., Татарников А. О. Огляд систем управління електронним навчанням. Матеріали МЦНД, 2023. С. 131-136.
31. Nafidi Y., El Batri B. A Comparative Analysis of Moodle and Google Classroom. 2023. P. 112-118.
32. Використання Microsoft Teams у дистанційному навчанні. Інформаційні технології і засоби навчання. 2024. С. 6-8.
33. Systematic Comparative Review of Three LMS Platforms: Moodle, Blackboard and Canvas. 2024. P. 48-50.
34. Spaska A., Kozub H., Abylasynova G., Kozub V., Koval Yu. Evaluation of innovative teaching methods using modern information technologies. Jurnal Ilmiah Ilmu Terapan Universitas Jambi. 2025. Vol. 9(1). P. 422-440.

35. Козуб Г. О., Шинкаренко Я. М., Козуб В. Ю. Гейміфікація в освіті: інтеграція Classcraft в навчальний процес. Педагогічна Академія: наукові записки. 2024. № 7. С. 20-29.
36. Гаврілова Л. М., Катасонова Ю. О. Теоретичні аспекти впровадження дистанційного навчання в Україні. 2017. С. 168-177.
37. Рамка цифрової компетентності педагогічних працівників : методичні рекомендації. МОН України. Київ, 2021-2022.
38. ISO/IEC 27001:2022. Information Security Management Systems - Requirements. Geneva, 2022.
39. ISO/IEC 11179-1:2023. Information technology - Metadata registries (MDR)
40. Закон України «Про вищу освіту»: № 1556-VII від 01.07.2014. URL: <https://zakon.rada.gov.ua/laws/show/1556-18>
41. Положення про дистанційне навчання: наказ МОН України № 466 від 25.04.2013 (зі змінами). URL: <https://zakon.rada.gov.ua/laws/show/z0703-13>
42. ДСТУ 3008:2015. Інформація та документація. Звіти у сфері науки і техніки. Київ : ДП «УкрНДНЦ», 2016
43. Deterding S., Dixon D., Khaled R., Nacke L. From Game Design Elements to Gamefulness: Defining “Gamification”. Proceedings of the 15th International Academic MindTrek Conference. 2011. P.9-15.
44. Huang B., Hew K. F. Do Points, Badges and Leaderboards Increase Learning and Activity? A Meta-Analysis. Educational Research Review. 2018. Vol. 24. P. 149-170.
45. Kremen V. H., Bykov V. Yu., Lytvynova S. H. Scientific and Methodological Support for Education Digitalization in Ukraine. Теорія і практика управління соціальними системами. 2022. № 1. С. 3-18.
46. Spaska A., Kozub H., Abylasynova G., Kozub V., Koval Yu. Evaluation of Innovative Teaching Methods Using Modern Information Technologies. Jurnal Ilmiah Ilmu Terapan Universitas Jambi. 2025. Vol. 9(1). P. 422-440.
47. Дехтярьова О. О., Літвінова А. М. Дослідження показників здоров'я в системі змішаного навчання // Вісник Луганського національного університету імені

- Тараса Шевченка. Педагогічні науки. 2021. № 1 (339), Ч. I. С. 115-126. DOI: 10.12958/2227-2844-2021-1(339)-1-115-126.
48. Морозова М. В. Інтерактивність як чинник ефективності дистанційного курсу // Вісник Луганського національного університету імені Тараса Шевченка. Педагогічні науки. 2021. № 6 (344), Ч. II. С. 123-132.
49. Собченко Т. М. Цифрова компетентність педагогів у системі дистанційної освіти // Вісник Луганського національного університету імені Тараса Шевченка. Педагогічні науки. 2021. № 8 (346), Ч. II. С. 157-166. DOI: 10.12958/2227-2844-2021-8(346)-2-157-166.
50. Караман О. Л. Інформаційно-цифрова компетентність як засіб комунікації учасників освітнього процесу // Вісник Луганського національного університету імені Тараса Шевченка. Педагогічні науки. 2022. № 1 (349), Ч. II. С. 26-33. DOI: 10.12958/2227-2844-2022-1(349)-2-26-33.
51. Биков В. Ю. *Моделі організаційних систем відкритої освіти*. Київ : АПН України, 2009. 684 с.
52. Van Steen M., Tanenbaum A. S. Distributed Systems. 3rd ed. 2017. [Електронний ресурс]. - Режим доступу: <https://www.distributed-systems.net/index.php/books/ds3/> (дата звернення: 24.11.2025).
53. Fielding R., Nottingham M., Reschke J. HTTP Semantics (RFC 9110). 2022. [Електронний ресурс]. - Режим доступу: <https://datatracker.ietf.org/doc/html/rfc9110> (дата звернення: 24.11.2025).
54. Bray T. The JavaScript Object Notation (JSON) Data Interchange Format (RFC 8259). 2017. [Електронний ресурс]. - Режим доступу: <https://www.rfc-editor.org/info/rfc8259> (дата звернення: 27.11.2025).
55. Laravel Documentation. Blade Templates (Laravel 12.x). [Електронний ресурс]. - Режим доступу: <https://laravel.com/docs/12.x/blade> (дата звернення: 02.12.2025).
56. Laravel Documentation. Eloquent: Getting Started (Laravel 12.x). [Електронний ресурс]. - Режим доступу: <https://laravel.com/docs/12.x/eloquent> (дата звернення: 02.12.2025).



57. PHP Manual. PDO::prepare. [Электронный ресурс]. - Режим доступа: <https://www.php.net/manual/en/pdo.prepare.php> (дата звернения: 02.12.2025).
58. MySQL 8.0 Reference Manual. Oracle. 2025. [Электронный ресурс]. - Режим доступа: <https://downloads.mysql.com/docs/refman-8.0-en.pdf> (дата звернения: 02.12.2025).
59. MySQL Documentation. FOREIGN KEY Constraints. [Электронный ресурс]. - Режим доступа: <https://dev.mysql.com/doc/en/create-table-foreign-keys.html> (дата звернения: 02.12.2025).
60. ISO. ISO/IEC 27001:2022 - Information security management systems - Requirements. [Электронный ресурс]. - Режим доступа: <https://www.iso.org/standard/27001> (дата звернения: 02.12.2025).
61. MoodleDocs. Moodle app offline features. [Электронный ресурс]. - Режим доступа: [https://docs.moodle.org/en/Moodle\\_app\\_offline\\_features](https://docs.moodle.org/en/Moodle_app_offline_features) (дата звернения: 08.12.2025).
62. Google for Education Help. Use Google Classroom offline. [Электронный ресурс]. - Режим доступа: <https://support.google.com/edu/classroom/answer/11015503> (дата звернения: 08.12.2025).
63. Sandhu R. S., Coyne E. J., Feinstein H. L., Youman C. E. Role-Based Access Control Models. IEEE Computer. 1996. [Электронный ресурс]. - Режим доступа: <https://csrc.nist.gov/csrc/media/projects/role-based-access-control/documents/sandhu96.pdf> (дата звернения: 08.12.2025).



Рис. А. 1. ER-Діаграма - структура бази даних.