

Київський столичний університет імені Бориса Грінченка
Факультет інформаційних технологій та математики
Кафедра комп'ютерних наук

Допущено до захисту

Завідувач кафедри комп'ютерних наук,
кандидат технічних наук, доцент

_____Ірина МАШКІНА

« ____ » _____ 2025 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня «бакалавр»

Спеціальність 122 Комп'ютерні науки

Освітня програма 122.00.01 Інформатика

**Тема: РОЗРОБКА ВЕБ-ПЛАТФОРМИ ДЛЯ ТЕСТУВАННЯ ЗНАНЬ
УЧНІВ У НАВЧАЛЬНИХ ЦІЛЯХ**

Виконав

студент групи ІНб-1-21-4.0д

Хохлов Дмитро Андрійович

(підпис)

Науковий керівник

кандидат педагогічних наук,

доцент кафедри комп'ютерних наук

Негоденко Олена Василівна

(підпис)

Київський столичний університет імені Бориса Грінченка
Факультет інформаційних технологій та математики
Кафедра комп'ютерних наук

«Затверджую»

Завідувач кафедри комп'ютерних
наук, кандидат технічних наук,
доцент

_____Ірина МАШКІНА

« ____ » _____ 2025р.

ЗАВДАННЯ НА КВАЛІФІКАЦІНУ РОБОТУ БАКАЛАВРА

«Розробка веб-платформи для тестування знань учнів у навчальних цілях»

Виконавець – студент групи

ІНБ-1-21-4.0д, спеціальності 122

Комп'ютерні науки, освітньої програми

122.00.01 Інформатика

Хохлов Дмитро Андрійович

1. Вихідні дані: Публікації з тематики електронного навчання, онлайн-тестування, веб-розробки освітніх платформ, принципів UX/UI-дизайну у сфері EdTech та стандарти розробки інтерфейсів для онлайн-освіти.
2. Основні завдання: Здійснити аналіз наукових та прикладних джерел за напрямками електронного тестування, веб-розробки освітніх систем і платформ для перевірки знань. Обґрунтувати мету, об'єкт, предмет і наукові засади дослідження.

Сформулювати структуру та зміст кваліфікаційної роботи.

Розробити архітектуру веб-платформи для створення, проходження та перевірки тестів. Реалізувати модулі управління користувачами, створення тестів, обробки відповідей, збереження результатів і формування статистики.

Створити інтуїтивно зрозумілий інтерфейс для учнів і викладачів.

Підготувати тези доповіді за результатами дипломної роботи.

Пояснювальна записка: Обсяг – до 48 сторінок формату А4 комп'ютерного набору з дотриманням вимог державного стандарту і методичних рекомендацій кафедри.

3. Графічні матеріали:

Презентація, що демонструє ключовий функціонал платформи.

4. Додатки:

Не передбачено.

5. Строк подання роботи на кафедру: «___» червня 2025 р.

Науковий керівник

к.п.н., доцент кафедри комп'ютерних
наук

_____ Негоденко О.В.

_____ дата

Виконавець:

_____ Хохлов Д.А.

_____ дата

Календарний план

№	Етап виконання роботи	Термін виконання	Примітка
1	Вибір теми дипломної роботи бакалавра та її затвердження	07.10.2024 – 10.10.2024	виконано
2	Аналіз існуючих платформ та технологій EdTech, опрацювання літератури та наукових статей	28.11.2024 – 13.12.2024	виконано
3	Постановка задачі, визначення об'єкта, предмета і мети дослідження	10.12.2024 – 20.12.2024	виконано
4	Вибір архітектури та методів зберігання даних, мов програмування для реалізації поставленого завдання	13.02.2025 – 25.02.2025	виконано
5	Складання структури системи, вимог до функціоналу, проєктування модулів	26.02.2025 – 09.03.2025	виконано
6	Створення серверної логіки: аутентифікація, тестування, збереження результатів	15.03.2025 – 26.03.2025	виконано
7	Розробка інтерфейсу користувача: створення та проходження тестів, перегляд результатів	01.04.2025 – 13.04.2025	виконано
8	Поеднання модулів, перевірка роботи системи, усунення помилок	14.04.2025 – 26.04.2025	виконано
9	Проведення аналізу ефективності платформи, узагальнення	28.04.2025 – 01.05.2025	
10	Оцінка результатів, підготовка висновків	04.05.2025 – 08.05.2025	виконано
11	Передзахист кваліфікаційної роботи	20.05.2025	виконано
12	Внесення правок, затвердження роботи, підготовка матеріалів для захисту	28.05.2025 – 10.06.2025	виконано
13	Захист кваліфікаційної роботи	17.06.2025	виконано

Здобувач

студент ІНБ 1-21-4.0д

Хохлов Дмитро Андрійович

Керівник роботи

к.п.н., доцент кафедри комп'ютерних наук

Негоденко Олена Василівна

РЕФЕРАТ бакалаврської роботи

Кваліфікаційна робота: 48 с., 27 рис., 16 посилань

Актуальність: На сьогодні, коли освіта все більше і більше переходить до використання цифрових технологій і потреба в дистанційному навчанні невинно зростає, розробка ефективних інструментів оцінювання знань стає надзвичайно актуальною. Онлайн-платформи для тестування набули статусу ключового елемента освітніх технологій, пропонуючи зручність, легкість масштабування та автоматизацію процесу контролю засвоєного матеріалу. Створення сучасної веб-платформи, яка дає можливість конструювати, проходити та аналізувати навчальні тести, допоможе суттєво покращити якість освіти та зробити навчальний процес приємнішим.

Об'єкт дослідження: Освітні платформи для організації тестування знань учнів.

Предмет дослідження: Технології та інструменти для реалізації веб-застосунків, що забезпечують функціональність створення, проходження тестів та аналізу результатів.

Мета роботи: Розробка веб-платформи для створення й проведення тестування учнів з можливістю перегляду результатів і статистики, орієнтованої на використання в навчальних закладах.

Завдання:

- провести аналіз існуючих платформ і рішень у сфері онлайн-тестування.
- обґрунтувати вибір архітектури системи та технологій реалізації.
- спроектувати і реалізувати модулі платформи: створення тестів, проходження тестування, облік результатів.
- забезпечити зберігання даних про тести та результати учнів у базі даних із доступом через API.

– створити зручний веб-інтерфейс для викладачів та учнів з урахуванням принципів UX/UI.

Основні результати: Розроблено веб-платформу, що дозволяє створювати навчальні тести, проходити їх та переглядати результати. Реалізовано повнофункціональний бекенд з API для управління даними та фронтенд-інтерфейс з підтримкою авторизації користувачів. Система протестована на предмет коректності виконання завдань та відображення результатів.

Практичне значення дослідження: Розроблена система може бути впроваджена в освітніх установах для автоматизації процесу оцінювання знань. Вона може застосовуватись у школах, коледжах, університетах або для внутрішнього навчання співробітників в організаціях.

Ключові слова: EdTech, тестування знань, веб-платформа, API, UX/UI, PHP, MySQL, phpMyAdmin, J-Query, Bootstrap, оцінювання, інтерактивне навчання, статистика результатів.

Зміст	
ВСТУП	8
1. ЗАГАЛЬНІ ПОЛОЖЕННЯ	10
1.1. Опис предметного середовища	10
1.2. Огляд наявних аналогів	15
1.3. Постановка задачі	20
1.4. Висновки до розділу	21
2. ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ	23
2.1. Вхідні дані	23
2.2. Вихідні дані	24
2.3. Опис структури бази даних	24
2.4. Висновки до розділу	27
2.5. Засоби розробки	28
2.6. Вимоги до апаратного забезпечення.	31
2.7. Висновки до розділу	33
3. ТЕХНОЛОГІЧНИЙ РОЗДІЛ	34
3.1. Керівництво адміністратора	34
3.2. Керівництво користувача	38
3.3. Керівництво програміста.	43
3.4. Текст програми.	44
3.5. Аналіз результатів роботи.	44
3.6. Висновки до розділу.	45
Висновки	46
ПЕРЕЛІК ПОСИЛАНЬ НА ДЖЕРЕЛА	48

ПЕРЕЛІК ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

JavaScript — динамічна, об'єктно-орієнтована, прототипна мова програмування.

jQuery — JavaScript-бібліотека.

СКБД — система управління базами даних.

AJAX — підхід до створення призначених для користувача інтерфейсів веб-застосунків, за якого вебсторінка, не перезавантажуючись, у фоновому режимі відсилає запити на сервер та сама звідти довантажує потрібні користувачу дані.

DOM — специфікація програмного інтерфейсу для роботи зі структурованими документами.

Відеотермінал — пристрій для віддаленого введення або виведення інформації в обчислювальних системах, оснащений екраном візуального контролю.

Відеомонітор — електронний пристрій для показу інформації. Пристрій, призначений для відтворення відеосигналу і візуального відображення інформації, отриманої від комп'ютера.

ВСТУП

У сучасному світі, де технологічний поступ стрімко набирає обертів, освіта також зазнає значних змін. Звичні методи навчання поступово витісняються цифровими інструментами, які пропонують більш гнучкий, індивідуалізований та зручний підхід до здобуття знань. Одним з ключових напрямків цифровізації освіти є впровадження електронних систем оцінювання, що дають змогу ефективно перевіряти знання, при цьому забезпечуючи прозорість, доступність та об'єктивність результатів. В цьому контексті розробка веб-платформи для складання тестів стала особливо важливою.

Актуальність цієї теми впливає з необхідності створення простих, масштабованих та універсальних інструментів для моніторингу знань студентів в освітньому процесі. Більшість існуючих сервісів або мають обмежений функціонал, або складний інтерфейс, що ускладнює їх використання у школах та вищих навчальних закладах. Водночас навчальним закладам потрібні інструменти, які не тільки дозволяють швидко створювати тести, але й забезпечують зручність для студентів під час їх складання та перегляду результатів.

Метою цієї дипломної роботи є створення веб-платформи для онлайн-тестування, яка відповідає освітнім потребам різних категорій користувачів – викладачів, студентів та адміністраторів. Платформа повинна забезпечувати простий механізм створення тестів, легкий доступ до проходження тестів з будь-якого пристрою, негайний зворотний зв'язок щодо результатів та зручний інтерфейс для перегляду оцінок через аналітику або звіти. Одним із ключових пріоритетів розробки є забезпечення інтуїтивно зрозумілого інтерфейсу та адаптивності системи до різних типів освітнього контенту.

Особливу увагу в цьому дослідженні приділено аспектам зручності використання веб-платформи. Чіткий та логічно організований інтерфейс користувача сприятиме швидкому засвоєнню функціоналу користувачами та знижує бар'єр входу для недосвідчених користувачів. Крім того, сучасні платформи повинні бути адаптованими до мобільних пристроїв, гнучкими та

підтримувати можливості персоналізації, що буде реалізовано у функціоналі розробленого застосунку.

Ще одним важливим компонентом проєкту є впровадження ефективної системи оцінювання. Платформа повинна не лише перевіряти правильність відповідей, але й зберігати історію спроб, аналізувати тенденції успішності учнів та надавати зворотний зв'язок викладачам для коригування методів викладання. Таким чином, тестування виконує не тільки контролюючу функцію, а й діагностичну, аналітичну та мотиваційну.

Відтак, створення веб-платформи тестування для освітніх цілей є актуальним кроком у напрямку модернізації освітнього процесу. Цей інструмент підвищить якість освіти, зробить її більш доступною, прозорою та технологічно орієнтованою. У рамках цієї дипломної роботи розробка, тестування та аналіз такої системи будуть виконані з урахуванням реальних потреб користувачів.

1. ЗАГАЛЬНІ ПОЛОЖЕННЯ

1.1. ОПИС ПРЕДМЕТНОГО СЕРЕДОВИЩА

1.1.1. Опис процесу діяльності

У сучасному світі освіта є фундаментом для особистісного, суспільного та загальнодержавного поступу.

В умовах стрімкого технічного прогресу, глобалізації та цифрової трансформації, освіта зобов'язана не лише відповідати вимогам сьогодення, але й бути інструментом адаптації людини до змін у навколишньому світі. Звідси випливає необхідність впровадження передових технологій в освітній процес, що допоможе підвищити ефективність навчання, розвивати навички самостійного отримання знань та гарантувати об'єктивну оцінку успішності учнів.

Актуальною проблемою в сфері освітніх технологій є розробка веб-платформ, які дають змогу автоматизувати перевірку знань та постійно контролювати успіхи учнів.

Цей підхід не тільки узгоджується з цілями цифрової трансформації освіти, але й вирішує ряд серйозних проблем сучасної освітньої системи, включно з суб'єктивним оцінюванням, відсутністю своєчасного зворотного зв'язку та складнощами аналізу результатів навчання протягом часу.

В контексті освітньої реформи в Україні та переходу до Нової української школи (НУШ), заснованої на компетентнісному підході, створення інноваційних інструментів для оцінювання знань є не тільки актуальним, а й стратегічно важливим. Сучасні школярі, що відносяться до поколінь Z та Alpha, ростуть в середовищі цифрових технологій, що вимагає нових підходів до представлення освітнього контенту та оцінювання знань. Веб-платформа, що дозволяє проводити інтерактивне тестування, адаптивний підбір завдань з урахуванням рівня підготовки студента, негайний аналіз результатів та рекомендації, є важливим кроком на шляху до формування якісного освітнього середовища.

Мета цього дослідження – розробити таку веб-платформу, яка поєднує функціональність для створення та проведення різних типів тестових завдань, збирання статистичних даних про результати та моніторинг успішності студентів з плином часу. У дослідженні будуть проаналізовані педагогічні, методологічні та технічні аспекти застосування автоматизованого тестування в навчальному процесі. Особлива увага приділятиметься побудові архітектури системи, забезпеченню надійного зберігання даних, організації ролей користувачів (учень, викладач, адміністратор) та запровадженню гнучких механізмів оцінювання знань.

Моніторинг академічних досягнень є ключовим компонентом забезпечення якості освіти. Традиційні методи оцінювання, як-от усні опитування або письмові іспити, мають ряд недоліків: вони трудомісткі, не дозволяють проводити аналіз великих обсягів даних в реальному часі та можуть бути суб'єктивними. Натомість автоматизоване тестування робить оцінювання ефективним, швидким, масштабованим та об'єктивним.

Платформа, описана в цій роботі, націлена на вирішення наступних завдань:

- Проведення індивідуального або масового тестування з відстеженням часу, оцінюванням та історією спроб;
- Генерація звітів про рівень засвоєння матеріалу;
- Візуалізація результатів за допомогою таблиць, діаграм та динамічних схем;
- Збереження історії оцінювання для аналізу прогресу учня з плином часу;
- Створення навчальних груп, курсів та тем, закріплених за конкретними вчителями або навчальними закладами;
- Адаптація складності завдань до рівня підготовки учня на основі попередніх результатів.

Така платформа буде корисна для освітніх закладів різних рівнів: початкових, середніх, позашкільних та навіть вищих. Вона буде ефективним інструментом не лише для вчителів, які зможуть бачити об'єктивну картину успішності учнів, а й для учнів, мотивуючи їх до самостійного навчання та

відповідальності за власні результати, а також для батьків, які отримують доступ до аналітики навчальних досягнень своєї дитини.

Даний проєкт стає особливо актуальним в умовах дистанційного та змішаного навчання, що набули широкого поширення внаслідок пандемії COVID-19. Потреба у швидкому зворотному зв'язку між учнем та викладачем та підтримці високої якості освіти в умовах фізичної розділеності стимулює впровадження нових цифрових рішень. Запропонована веб-платформа пропонує гнучкість, доступність та зручність з будь-якого пристрою, що значно підвищує її практичну цінність. Моніторинг успішності в запропонованому програмному рішенні буде реалізовано як безперервний процес збору та аналізу результатів тестування учнів, формування індивідуальних профілів досягнень та візуалізації динаміки знань з часом. Це дозволить виявляти проблемні теми, коригувати персоналізовані навчальні траєкторії, підвищувати мотивацію та об'єктивність в оцінюванні, що критично важливо як для учнів, так і для викладачів.

Загалом, розробка та впровадження веб-платформи для перевірки знань учнів та моніторингу успішності є важливим кроком до цифрової трансформації української освіти. Проєкт має як соціальне, так і практичне значення, оскільки його метою є покращення якості освіти, забезпечення прозорості навчального процесу та підтримка як вчителів, так і учнів у досягненні високих освітніх результатів. Успішна реалізація такого проєкту може слугувати зразком для масштабування на інші навчальні заклади та системи управління освітою.

1.1.2. Опис функціональної моделі

Основна мета розробки веб-платформи для оцінювання знань студентів в освітніх закладах та відстеження їх навчальних досягнень полягає у створенні зручного, гнучкого та ефективного інструменту, що автоматизує процеси оцінювання та аналізу навчальних результатів. Ця система повинна значно полегшити щоденну роботу усіх користувачів – студентів, викладачів та адміністраторів – надаючи їм необхідний функціонал для взаємодії з навчальним контентом та результатами тестування.

Для забезпечення комфортного та продуктивного користування платформою, важливо чітко визначити функціональні можливості, які потрібно реалізувати. В ідеалі, система має містити такі основні модулі:

- **Керування користувачами та ролями:** автентифікація та розмежування прав доступу для студентів, викладачів та адміністраторів. Цей компонент є ключовим для забезпечення безпеки та персоналізованої взаємодії з платформою.
- **Створення, редагування тестових завдань:** викладачі повинні мати змогу створювати різні типи тестів (з вибором однієї правильної відповіді, питання з відкритою відповіддю, завдання на відповідність та питання на декілька відповідей), адаптуючи їх до потреб певних груп студентів.
- **Призначення тестів групам або окремим студентам:** забезпечення гнучкості в навчальному процесі, що дозволяє проводити як загальні тести, так і індивідуальну оцінку знань.
- **Проведення онлайн-тестування з обмеженнями часу та кількості спроб:** це збільшує дисципліну студентів та запобігає багаторазовим, несанкціонованим спробам тестування.
- **Моніторинг активності та успішності студентів у режимі реального часу:** це дозволяє викладачам швидко реагувати на проблеми та корегувати освітні стратегії.
- **Адміністративний модуль для управління навчальними групами, курсами, предметами та користувачами:** цей блок підтримує організаційну структуру навчального процесу.

Реальний функціонал розробленої веб-платформи відповідає більшості цих вимог. Зареєстровані користувачі отримують доступ до інтерфейсів, що відповідають їх ролям. Викладачі можуть створювати, редагувати та видаляти різноманітні тестові завдання, призначати тести певним групам чи студентам, контролювати терміни виконання та переглядати звіти. Студенти можуть проходити призначені тести, переглядати свої оцінки та історію тестування.

Адміністратори управляють користувачами, групами та предметами, забезпечуючи повноту та актуальність інформації в системі.

Завдяки інтеграції автоматизованого оцінювання та візуалізації результатів, платформа забезпечує об'єктивність та швидкість оцінювання рівня знань, значно покращуючи якість зворотного зв'язку для студентів та викладачів. Збереження історії тестування дає змогу аналізувати тенденції успішності як на індивідуальному, так і на груповому рівнях.

Особлива увага приділяється зручності використання. Інтерфейс розроблено у сучасному, зрозумілому стилі та адаптовано для різних пристроїв, що дозволяє проводити тестування як у класах, так і дистанційно. Система забезпечує швидку навігацію між розділами, чіткі підказки та корисні інструменти для викладачів та учнів.

Обґрунтовуючи вибір функціональності, варто підкреслити, що в сучасних умовах цифрової трансформації освіти важливо забезпечити комплексне рішення, яке не тільки автоматизує рутинні операції, але й підвищує ефективність навчального процесу. Автоматизація тестування звільняє викладачів від трудомістких розрахунків балів, а аналітичні інструменти дозволяють глибше зрозуміти особливості успішності учнів. Адаптивний підхід до вибору завдань сприяє персоналізованому навчанню, що є ключовим напрямом у сучасній педагогіці.

Водночас слід зазначити, що розроблена система має потенціал для подальшого розширення функціональності, включаючи інтеграцію з електронними журналами оцінювання, впровадження додаткових типів завдань, складніших адаптивних алгоритмів та розширеної аналітики. Однак, навіть на даному етапі, платформа відповідає основним потребам учасників освітнього процесу, значно покращуючи якість організації та контролю знань.

Таким чином, функціональна модель веб-платформи для оцінювання знань та моніторингу успішності учнів є комплексною, орієнтованою на користувача та здатна ефективно підтримувати сучасний освітній процес у цифровому

середовищі. Більш детально і точно це зображено на діаграмі варіантів використання 1.1.

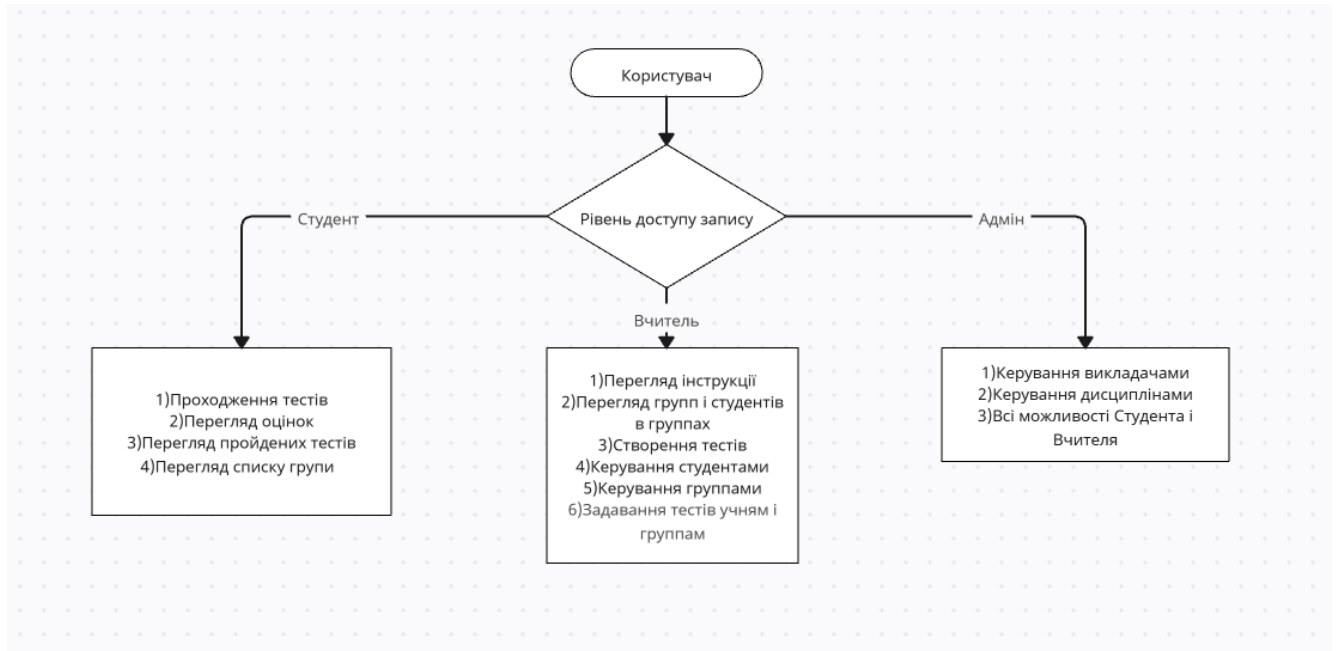


Рис. 1.1. Діаграма варіантів використання системи

Аналізуючи діаграму варіантів використання системи, можна зробити висновок, що цей програмний продукт є системою з універсальним набором функцій, які забезпечують якісну та ефективну взаємодію з користувачем, значно спрощуючи виконання повсякденних завдань.

1.2. ОГЛЯД НАЯВНИХ АНАЛОГІВ

У процесі дослідження було проаналізовано декілька популярних онлайн-платформ, які забезпечують можливість створення й проведення інтерактивних навчальних тестів. Зокрема, розглянуто сервіси **Kahoot**, **Blooket**, **Google Forms** та **ClassMarker**, які на сьогодні активно застосовуються в освітньому процесі різного рівня.

1. Kahoot

Kahoot — це онлайн-платформа для створення інтерактивних вікторин, яка має яскраво виражений ігровий формат. Вона орієнтована переважно на шкільну

та університетську аудиторію, але також використовується в корпоративному навчанні.

Основний функціонал:

- Створення вікторин, опитувань, "true/false" тестів, головоломок.
- Можливість додавання відео, зображень, музики.
- Проведення тестів у реальному часі (Live Mode) або асинхронно (Challenge Mode).
- Детальна аналітика результатів (у платній версії).
- Інтеграція з Google Classroom та Microsoft Teams.

Переваги:

- Висока інтерактивність та гейміфікація: відповіді супроводжуються яскравими анімаціями та звуками.
- Миттєвий зворотній зв'язок та видимість результатів для викладача.
- Простий, інтуїтивно зрозумілий інтерфейс.
- Працює на будь-яких пристроях: ПК, планшетах, смартфонах.

Недоліки:

- Лише обмежена кількість типів запитань.
- Немає можливості налаштувати логіку проходження тесту (наприклад, перехід на запитання залежно від відповіді).
- Не підтримує введення довгих письмових відповідей.
- Складно застосовувати для формального контролю знань, особливо у віддаленому форматі.
- У безкоштовній версії — обмеження на кількість учасників та базова аналітика.

Практичне застосування:

Kahoot активно використовується для проведення швидких опитувань, повторення матеріалу перед тестами, неформального закріплення знань у класі. Часто застосовується як "розігрів" перед основним заняттям(рисунок 1.2.).

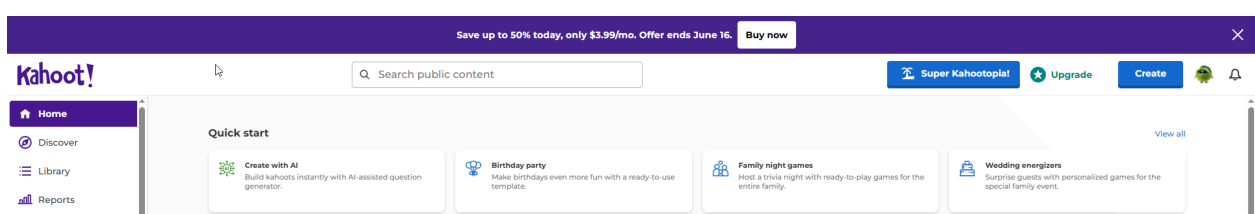


Рис. 1.2. Зовнішній вигляд інформаційної сторінки сайту Kahoot.com

2. Google forms

Google Forms — це універсальний сервіс від Google для створення онлайн-форм, який часто використовується для анкетування, збору даних та створення тестів із самоперевіркою. (рисунок 1.3.).

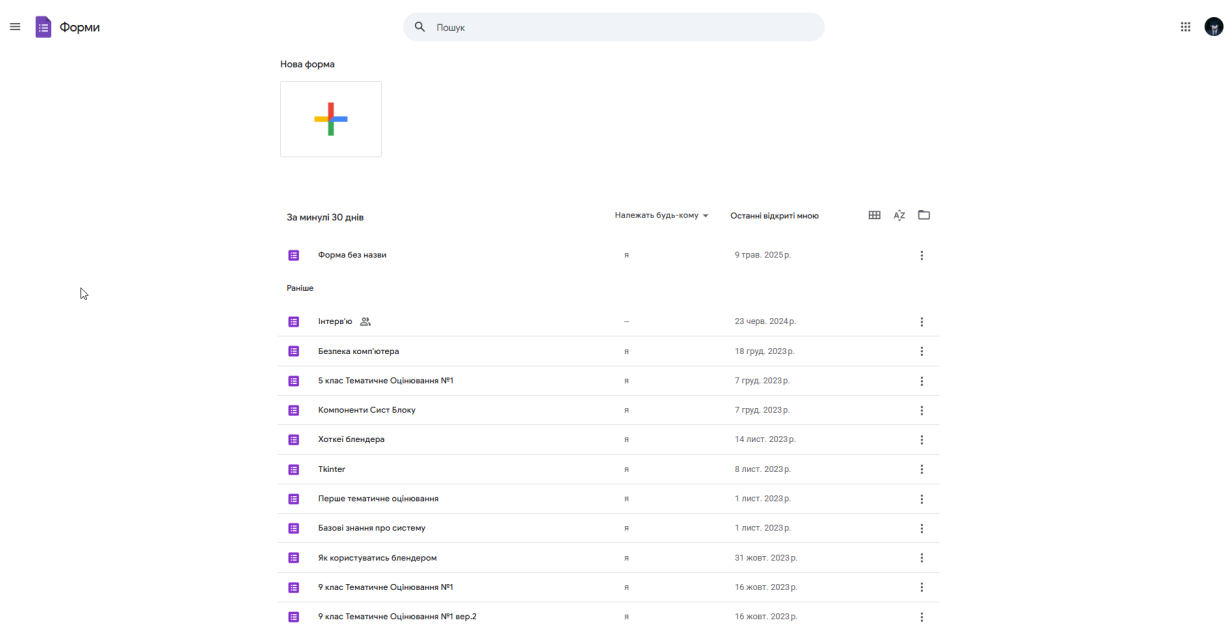


Рис. 1.3. Інформаційна сторінка ресурсу

Google Forms дозволяє створювати тести з різними типами запитань, зокрема питання з вибором однієї або кількох відповідей, шкали, списки, а також текстові відповіді — як короткі, так і розгорнуті. У налаштуваннях можна активувати режим тестування з автоматичним оцінюванням, що значно полегшує перевірку відповідей. Додатково реалізована можливість створення логічних переходів між секціями форми, що дозволяє реалізувати базову адаптивність тесту залежно від відповідей користувача. Всі зібрані відповіді зберігаються в таблиці Google Sheets, що дає змогу здійснювати подальший аналіз і обробку даних. Google Forms не має ігрових елементів, однак завдяки своїй універсальності та

відкритості є зручним інструментом для масового опитування чи формального тестування.

Переваги:

- Повністю безкоштовний.
- Інтеграція з Google Workspace (Docs, Drive, Classroom).
- Гнучкість у створенні форм і типах запитань.
- Можна створити форму з довільною кількістю запитань, необмеженою кількістю респондентів.

Недоліки:

- Відсутність гейміфікації.
- Базовий, одноманітний інтерфейс (неможливо оформити у стилі гри або мобільного додатку).
- Немає захисту від списування.
- Мала кількість типів завдань для оцінювання навичок (немає перетягування, сортування тощо).

3. *Blooket*

Blooket — це платформа, яка поєднує традиційне тестування з відеоігровими механіками. Вона особливо популярна серед учнів молодшої та середньої школи завдяки яскравому візуальному оформленню, цікавим форматам проходження тестів і системі винагород.

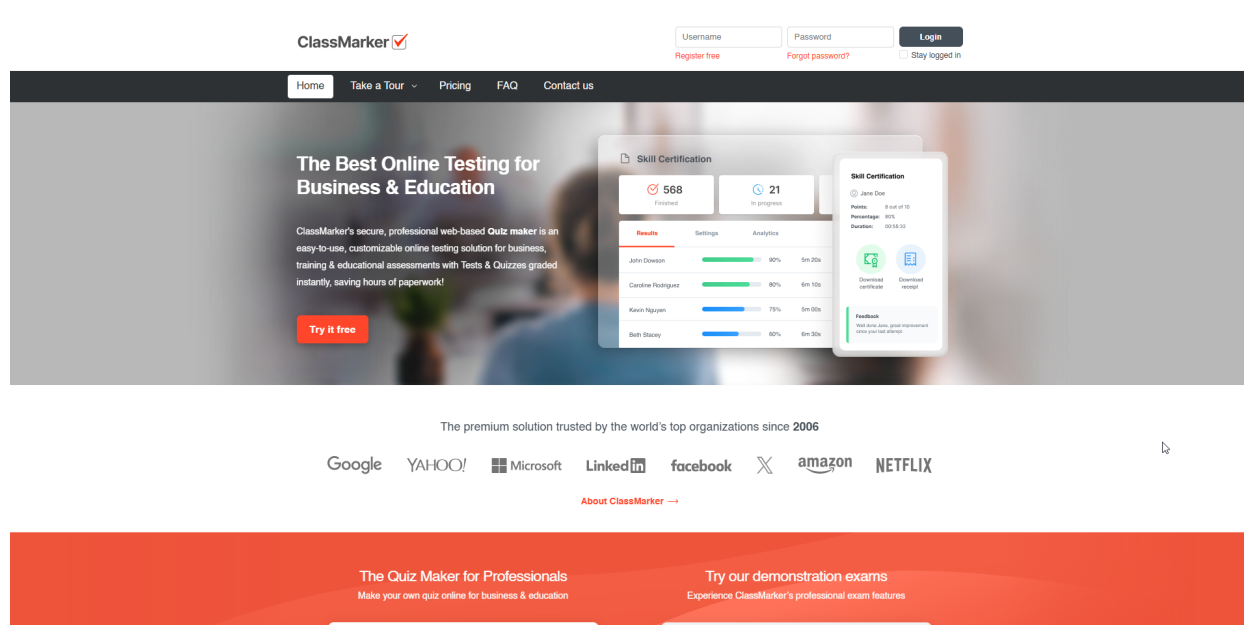
Основний функціонал:

Сервіс дозволяє створювати набори запитань з варіантами відповідей або імпортувати їх із зовнішніх платформ, таких як Quizlet. Кожен набір тестових запитань може бути використаний у різних ігрових режимах — наприклад, у вигляді стратегій, аркад, економічних симуляцій або змагань між учасниками. Учні можуть проходити ці тести індивідуально або в командах. За правильні відповіді гравці отримують внутрішньоігрові нагороди — персонажів або ігрові валюти, що стимулює інтерес до навчання. Платформа також підтримує як

синхронний, так і асинхронний режим проходження. Незважаючи на ігровий формат, Blooket зберігає базову аналітику відповідей, яка може бути корисною для оцінювання загального рівня засвоєння знань.

4. *Classmaker*

ClassMarker підтримує створення тестів із різноманітними типами запитань, серед яких: вибір правильної відповіді, запитання на встановлення відповідності,



запитання з відкритою відповіддю, есе, запитання з сортуванням. Платформа дає змогу встановлювати часові обмеження на виконання тесту, захищати доступ паролями, обмежувати кількість спроб, фільтрувати IP-адреси та застосовувати інші інструменти безпеки. Для кожного користувача можна індивідуально налаштувати умови проходження тесту. За результатами тестування система автоматично генерує звіти та може видавати сертифікати. Статистику можна переглядати у вигляді таблиць, діаграм або експортувати для подальшого аналізу. Хоча ClassMarker не має гейміфікаційних елементів, він вирізняється високим рівнем надійності, деталізованої аналітики та підходить для формального оцінювання знань у будь-яких навчальних або бізнес-середовищах.

Рис. 1.4- Основна сторінка програми «Журнал преподавателя» ProSoft-Design

1.3. ПОСТАНОВКА ЗАДАЧІ

1.3.1. Призначення розробки

Основною метою розробки веб-платформи для перевірки знань студентів та моніторингу успішності є створення сучасного, гнучкого та зручного інструменту, який автоматизує процес оцінювання в навчальних закладах. Ця платформа розроблена для обслуговування студентів, викладачів та адміністраторів, забезпечуючи централізоване середовище, де тести можна створювати, призначати та виконувати онлайн, одночасно забезпечуючи моніторинг та аналіз успішності студентів у режимі реального часу.

Автоматизуючи рутинні завдання, такі як оцінювання та створення звітів, система прагне зменшити ручне навантаження, підвищити точність та забезпечити своєчасний зворотний зв'язок усім зацікавленим сторонам. Зрештою, платформа має підвищити якість освіти, підтримуючи прийняття рішень на основі даних та персоналізовані навчальні шляхи.

1.3.2. Цілі та задачі розробки

Основною метою цього проекту є створення інтегрованого рішення, що автоматизує процеси створення, адміністрування та оцінювання тестів, а також ефективний моніторинг навчальних досягнень студентів протягом навчального року.

Для досягнення цієї мети платформа має включати ергономічний та інтуїтивно зрозумілий інтерфейс, адаптований до потреб різних користувачів: студентів, викладачів та адміністраторів.

Ключові цілі включають реалізацію безпечної аутентифікації користувачів та управління ролями, надання викладачам можливості створювати різні типи тестів та призначати їх групам або окремим студентам, а також забезпечення автоматичного підрахунку оцінок.

Крім того, система повинна надавати комплексні аналітичні інструменти, такі як звіти про успішність та візуальне представлення даних, щоб допомогти викладачам виявляти слабкі місця в навчанні та відповідно коригувати навчальний процес.

Забезпечення конфіденційності даних та надійності системи також залишається першочерговим завданням у процесі розробки.

1.4. Висновки до розділу

Внаслідок дослідження функціональної моделі, мети та задач розробки веб-платформи для оцінювання знань студентів та контролю успішності, можна підсумувати, що розроблена система відповідає сучасним тенденціям цифровізації освітнього процесу. Вона гарантує результативне виконання ключових функцій: автоматизацію тестування, моніторинг успішності, аналіз досягнень та надання оперативної зворотної інформації.

Платформа надає всебічну підтримку ролям студентів, викладачів та адміністраторів, пропонуючи кожній категорії користувачів зручний та інтуїтивно зрозумілий інтерфейс. Основні переваги включають у себе: можливість створення тестів з різними форматами питань, керування правами доступу, призначення тестів для груп або індивідуальних студентів, формування звітів та візуалізацію результатів успішності.

Отже, запропонована система забезпечує об'єктивне та прозоре оцінювання знань, підтримує індивідуалізований підхід до навчання, оптимізує робоче навантаження викладачів та дає змогу вчасно виявляти труднощі у навчанні для подальшого вдосконалення освітнього процесу.

2. ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ

2.1. Вхідні дані

Вхідні дані надходять від користувачів системи до адміністратора, який заповнює БД, після надходження заявки на реєстрацію.

Дані що оброблюються адміністратором

- ПІБ викладача;
- логін викладача;
- пароль викладача;
- дисципліна, яку викладає викладач
- ПІБ студента;
- логін користувача;
- пароль користувача

Дані що оброблюються у БД

- ПІБ викладача;
- ПІБ студента;
- тип користувача;
- пароль користувача;
- назва дисципліни;
- назва групи;
- оцінка;
- перелік дисциплін у групи;
- перелік студентів, що належать до групи;
- перелік викладачів групи;
- перелік дисциплін, що викладає викладач;

2.2. Вихідні дані

Дані, що оброблюються модулем поточної успішності студентів

- підсумкова оцінка;
- семестрова оцінка;
- список викладачів;
- список груп;
- список студентів;
- список дисциплін;
- список оцінок;

2.3. Опис структури бази даних

Безумовно, розробка інформаційної системи для організації оцінювання студентів потребує детального дослідження потреб користувачів та специфіки предметної сфери. На відміну від простої системи оцінювання, ця база даних забезпечує повний спектр функціональності – від формування тестів та питань до збереження спроб, відповідей, балів та результатів.

У процесі проєктування враховувалися взаємодії між студентами, викладачами та адміністраторами, що сприяло створенню гнучкої та масштабованої архітектури. База даних реалізована з використанням реляційної моделі, яка гарантує високий рівень цілісності даних та допомагає уникнути їх дублювання.

В результаті аналізу було визначено основні сутності: користувачі, групи, предмети, тести, питання, варіанти відповідей, спроби, відповіді, тестові завдання та коректні відповідності. Ці сутності та їх зв'язки дозволяють ефективно організовувати навчальний процес та збирати аналітику успішності студентів.

Ключовою таблицею є **"users"** ("користувачі"), яка містить записи для всіх системних ролей: адміністратори, викладачі та студенти. Поле **"status"** визначає роль користувача в системі. Кожен студент належить до певної

групи та може мати пов'язані оцінки та основний предмет. Таблиця **"groups"** містить назви академічних груп та інформацію про предмети, які вони вивчають.

Академічні дисципліни описані в таблиці **"subjects"**, яка зберігає назву предмета та ідентифікатор викладача, що за нього відповідає. Це дає змогу відслідковувати, який викладач закріплений за яким курсом.

Особливу увагу приділено функціональності тестування. Таблиця **"tests"** містить назви тестів. Кожен тест містить перелік питань, які мають тип (один/багато варіантів, на відповідність, відкриті тощо), текст питання та кількість балів. Кожне питання може містити декілька варіантів відповідей, що зберігаються в таблиці **"options"** ("варіанти"). Для питань на відповідність використовується додаткова таблиця **"correct_matches"** для визначення пар правильних відповідей.

Користувачі проходять тести, і кожна спроба записується в таблицю **"attempts"** ("спроби") з міткою часу, балом та посиланнями на користувача та тест. Фактичні відповіді (включаючи вибрані варіанти, письмові відповіді або пари відповідностей) зберігаються в таблиці **"answers"** ("відповіді").

Для організації доступу до тестів використовується таблиця **"users_tests"**. Вона дозволяє призначати тести певним користувачам або групам з обмеженням часу. Це надає викладачам інструменти для контролю знань під час занять або домашніх завдань.

Вся архітектура бази даних логічно структурована для уникнення надмірності, забезпечення цілісності даних та дотримання основних принципів нормалізації. На основі проаналізованих зв'язків та сутностей було створено відповідну ER-діаграму, яка представлена на рисунку 2.2.

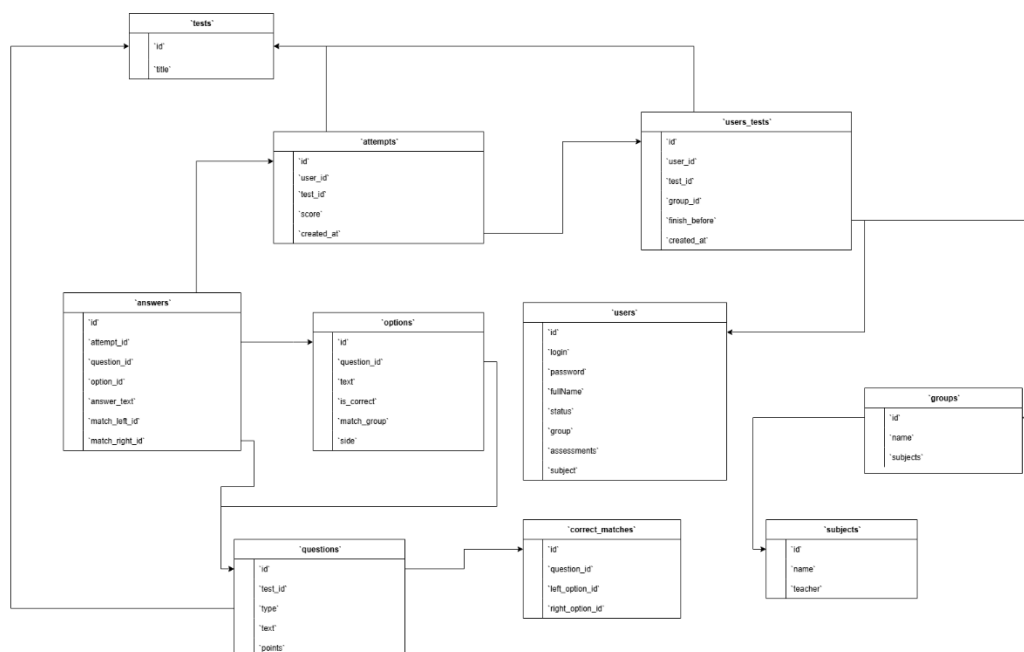


Рис. 2.1. Макет бази даних.

Було обрано RHPMyAdmin як систему керування базами даних, оскільки він простий у використанні, має широку доступність та підтримку. Інтерфейс RHPMyAdmin полегшує швидке виконання ключових операцій з базами даних, включно зі створенням, редагуванням, переглядом, видаленням, а також імпортом та експортом таблиць — і все це можливе в кілька кліків. Це суттєво прискорює процес розробки та тестування.

Важливою особливістю є можливість експорту дампа бази даних, яка дає змогу розробникам створювати резервні копії всієї структури та даних. Ця функція надзвичайно корисна при міграції платформи на інший хост або під час створення контрольних точок версій у процесі розробки. За необхідності дамп можна зберігати у стисненому форматі, що робить його зручним для зберігання та передачі. Простота та інтуїтивно зрозумілий інтерфейс RHPMyAdmin зробили його одним з найпопулярніших інструментів серед веб-розробників.

RHPMyAdmin зазвичай включений у серверну інфраструктуру, що надається більшістю хостинг-провайдерів, і також доступний у локальних середовищах розробки, наприклад, XAMPP, Denwer чи OpenServer. Це дає змогу розробникам працювати з базою даних локально, не потребуючи

постійного підключення до Інтернету, що є особливо корисним на ранніх етапах розробки та тестування.

Структура даних платформи базується на SQL (Structured Query Language) — стандартній мові запитів для взаємодії з реляційними базами даних. SQL застосовується для створення таблиць, визначення зв'язків між ними та управління даними через вставку, оновлення та отримання записів. Це забезпечує ефективну обробку запитів навіть при роботі з великими обсягами даних, що критично важливо для зберігання великої кількості тестів, питань, спроб та відповідей.

SQL розроблено для роботи з реляційними базами даних — структурованими системами, де дані організовані у взаємопов'язані таблиці. Такий підхід забезпечує цілісність даних, уникнення надмірності та масштабованість системи для задоволення потреб різних навчальних закладів.

Отже, поєднання PHPMyAdmin як інструменту управління та SQL як основи структури даних забезпечує надійний, зрозумілий та зручний механізм для розробки повнофункціональної веб-платформи для освітнього тестування.

2.4. Висновки до розділу

В даному розділі було позначено інформаційне забезпечення при У цьому розділі досліджується інформаційне підґрунтя, задіяне у розробці веб-платформи для освітнього тестування.

Було встановлено, що створення повної бази даних є критичним етапом для правильної роботи такої системи, з огляду на обробку великого обсягу даних, які надходять від користувачів. Додатково було обрано зручну та поширену систему управління базами даних, щоб гарантувати ефективне керування структурою та вмістом бази даних на всіх етапах розробки.

2.5. ЗАСОБИ РОЗРОБКИ

2.5.1. Середовище розробки

Під час розробки системи було використано різноманітні сучасні програмні засоби для забезпечення ефективного впровадження та зручного тестування. Ключові інструменти включали:

- текстовий редактор **VSCode**;
- серверне середовище **XAMPP** — безкоштовний кросплатформний пакет з відкритим кодом;
- система керування базами даних **PHPMyAdmin**;
- серверну мову сценаріїв **PHP**;
- **HTML** для структурування сторінок;
- **CSS** для стилізації інтерфейсу;
- **SQL** для роботи з базою даних;
- **jQuery** — бібліотеку JavaScript для покращення інтерактивності;
- **Bootstrap** — адаптивний фреймворк для веб-дизайну.

VSCode — це потужний кросплатформний текстовий редактор, що забезпечує підтримку широкого спектру мов програмування, таких як C, C++, C#, CSS, HTML, JavaScript та PHP. Він характеризується підсвічуванням синтаксису, високою продуктивністю та розширеною системою плагінів, більшість з яких розроблена на Python. Незважаючи на те, що він не є повністю відкритим або безкоштовним програмним забезпеченням, він може бути використаний без будь-яких обмежень, а його функціональність значно розширюється за допомогою екосистеми плагінів, підтримку якої здійснюють розробники.

PHPMyAdmin — це широкоживане веб-застосування з графічним інтерфейсом для управління базами даних MySQL або MariaDB. Воно дозволяє користувачам здійснювати SQL-запити, переглядати та змінювати вміст таблиць, а також створювати резервні копії баз даних безпосередньо через браузер. Простота

та доступність зробили його одним з найпопулярніших інструментів для адміністрування баз даних, що часто надаються хостинговими сервісами.

XAMPP — це кросплатформне середовище розробки, яке містить Apache, MariaDB/MySQL, PHP, Perl та додаткові компоненти, такі як OpenSSL та PHPMyAdmin. Воно надає можливість швидкого розгортання локального сервера для цілей тестування та розробки. Завдяки інтуїтивно зрозумілому інтерфейсу та легкості налаштування, XAMPP набув популярності серед новачків та досвідчених розробників.

2.5.2. Опис мови

Для розробки веб-платформи для освітнього тестування було задіяно різні мови програмування та технології, аби забезпечити функціональну повноту, гнучкість та ефективність системи. Головний технологічний стек охоплює PHP, HTML, CSS, SQL, jQuery та Bootstrap. Кожна з них має своє призначення в архітектурі програмного продукту.

PHP — це мова скриптів на стороні сервера, яка використовується у веб-розробці[1], яка призначена для динамічної генерації HTML-сторінок. Вона є однією з найширше використовуваних мов у веб-розробці, завдяки своїй простоті, великій функціональності та підтримці від більшості хостинг-провайдерів. PHP-код виконується на сервері, і клієнту надсилається лише результуюча HTML-розмітка, що покращує безпеку.

PHP дозволяє вбудовувати HTML безпосередньо у свій код та може динамічно генерувати JavaScript. Підтримує об'єктно-орієнтоване програмування, зокрема класи, інтерфейси, спадкування та обробку винятків. Мова значно розвинулась з введенням потужної об'єктно-орієнтованої моделі в PHP 5.0 (2004), анонімних функцій та просторів імен в PHP 5.3 (2009), а також функцій першого класу, які дозволяють програмування вищого порядку.

HTML (мова розмітки гіпертексту) — це стандартна мова розмітки, що застосовується для структурування веб-документів. Вона визначає заголовки,

абзаци, списки, форми, зображення та інші елементи інтерфейсу користувача. HTML є основою для веб-контенту та сприяє семантичному структуруванню інформації. У сучасній веб-розробці вона тісно взаємодіє з CSS та JavaScript для забезпечення динамічного та інтерактивного досвіду користувача.

CSS (каскадні таблиці стилів) — це мова таблиць стилів, яка використовується для визначення зовнішнього вигляду HTML-документів. Вона дозволяє розробникам задавати кольори, шрифти, поля, макети та інші параметри представлення, розділяючи контент та його відображення.

Таке розділення полегшує підтримку коду, забезпечує масштабованість та адаптивність інтерфейсів до різних пристроїв (моніторів, смартфонів, принтерів тощо).

SQL (структурована мова запитів) — це декларативна мова програмування, що використовується для взаємодії з реляційними базами даних. Вона застосовується для створення, оновлення, запитів та керування схемами баз даних і контролем доступу користувачів. SQL не є окремим програмним продуктом чи системою управління базами даних, а стандартною мовою для спілкування з такими системами. Вона дозволяє створювати інтерактивні запити та може бути вбудована в прикладні програми для керування даними.

jQuery - це бібліотека JavaScript з відкритим кодом[2], яка була створена з метою спростити та прискорити розробку веб-додатків, розроблена для спрощення маніпулювання елементами DOM, обробки подій, анімації та взаємодії AJAX. Вона високо оцінюється за лаконічний синтаксис, кросбраузерну сумісність та підтримку від спільноти.

jQuery покращує інтерактивність та адаптивність веб-сторінок, надаючи інструменти для створення динамічних інтерфейсів користувача. Основна функціональність включає повернення самого об'єкта, що дозволяє здійснювати ланцюжкове використання методів та оптимізувати програмування.

Bootstrap — це безкоштовний фронтенд-фреймворк з відкритим кодом, призначений для створення адаптивних та візуально привабливих веб-сайтів та

веб-додатків. Він містить попередньо створені шаблони HTML та CSS для типографіки, форм, кнопок, навігації, сповіщень тощо. До нього також включено компоненти JavaScript для модальних вікон, каруселей, випадаючих списків та інших інтерактивних елементів.

Bootstrap є модульним, в основному складається з таблиць стилів LESS, які розробники можуть налаштовувати відповідно до потреб проєкту.

Ключові компоненти Bootstrap включають:

- **Система сітки** — готові до використання стовпці.
- **Шаблони** — фіксовані або адаптивні шаблони сторінок.
- **Типографіка** — класи для різних шрифтів, включаючи стилі коду та цитат.
- **Медіа** — інструменти для керування зображеннями та відео.
- **Таблиці** — інструменти форматування таблиць, зокрема функцію сортування.
- **Форми** — класи для розробки форм та пов'язаних з ними подій.
- **Навігація (меню, панель навігації)** — класи для вкладок, сторінок та меню.
- **Сповіщення** — класи для модальних вікон, підказок та спливаючих сповіщень.
- **Шрифти значків** — набір значків у форматі шрифту, що складається з майже 500 компонентів.

Колективне використання цих мов програмування та інструментів дало змогу створити веб-платформу, яка є функціонально повною, безпечною, зручною для користувача та адаптованою до різних технічних середовищ.

2.6. Вимоги до апаратного забезпечення.

Для повного використання системи підтримки моніторингу успішності студентів користувач повинен мати стабільне підключення до Інтернету та сучасний веб-браузер. Оскільки система реалізована як веб-платформа, всі її функції працюють онлайн, що вимагає постійного доступу до Інтернету з

пропускною здатністю щонайменше 1 Мбіт/с для забезпечення належного завантаження елементів інтерфейсу та взаємодії з базою даних у режимі реального часу.

Рекомендовані браузері для доступу до системи:

- Mozilla Firefox – стабільний та безпечний браузер з відкритим кодом, який підтримує сучасні веб-технології, включаючи HTML5, CSS3, JavaScript та WebAssembly.
- Google Chrome – один з найшвидших браузерів з широкою підтримкою найновіших веб-стандартів, частими оновленнями безпеки та вбудованими інструментами для розробників.
- Веб-браузер Opera – багатофункціональний браузер, що пропонує вбудований VPN, блокувальник реклами та підтримку динамічних веб-додатків.

Важливо, щоб користувачі використовували останню (актуальну) версію браузера, оскільки застарілі версії можуть не підтримувати певні скрипти, бібліотеки або стилі, що використовуються у веб-додатку.

Мінімальні системні вимоги до персонального комп'ютера користувача:

- Операційна система: Windows 7, Windows 8 або Windows 10 (рекомендується 64-розрядна версія для кращої сумісності та продуктивності).
- Процесор: центральний процесор з тактовою частотою не менше 1 ГГц (рекомендується двоядерний або багатоядерний 64-розрядний процесор).
- Оперативна пам'ять: щонайменше 4 ГБ (для одночасної роботи веб-браузера, системних процесів та фонових завдань; в ідеалі 8 ГБ, щоб уникнути затримки).

- Сховище: щонайменше 1 ГБ вільного місця на жорсткому диску або SSD (для кешу браузера, тимчасових файлів та належної роботи ОС).

2.7. Висновки до розділу

У даному розділі аргументується вибір інструментарію для розробки, що охоплює мови програмування PHP, HTML, CSS, SQL, JavaScript (з використанням бібліотеки jQuery) та фреймворк Bootstrap. Розглядаються функції, переваги та релевантність кожного компонента для створення освітньої веб-платформи. Акцентовується увага на мінімальних системних вимогах до софту та «заліза», необхідних для безперебійної та безпечної роботи системи. Встановлено, що користувачі матимуть доступ до платформи за наявності сучасного браузера, комп'ютера, що відповідає базовим специфікаціям, та стабільного інтернет-з'єднання. Комбінація обраних інструментів гарантує надійність, гнучкість та адаптивність розробленої системи.

3. ТЕХНОЛОГІЧНИЙ РОЗДІЛ

3.1. КЕРІВНИЦТВО АДМІНІСТРАТОРА

При загрузці сайту з'являється вікно авторизації, зображене на рисунку 3.1.

Рис. 3.1. Вікно авторизації сайту

Після входу в систему, користувач «Адміністратор» переходить на головну сторінку сайту, з навігаційною панеллю у верхній частині сторінки (рисунок 3.2.). Після натискання кнопки «Викладачі» - користувач переходить на сторінку зі списком усіх викладачів та формою створення нового (рисунок 3.3.).

Рис. 3.2. Навігаційна панель

#	ПІБ	Дисципліна	Логін	Пароль	
1	Стародуб Олександр Петрович		teacher1	321	Видалити
2	Тетерева Марина Ярославовна		teacher2	321	Видалити

Список викладачів

ПІБ

Логін

Пароль

Рис. 3.3. Сторінка управління викладачами

При натисканні на вкладку «Групи» - користувач «Адміністратор» переходить на іншу сторінку, на якій розташований список усіх груп, з кнопками «Дисципліни», «Видалити» та «Студенти» (рисунок 3.4.).

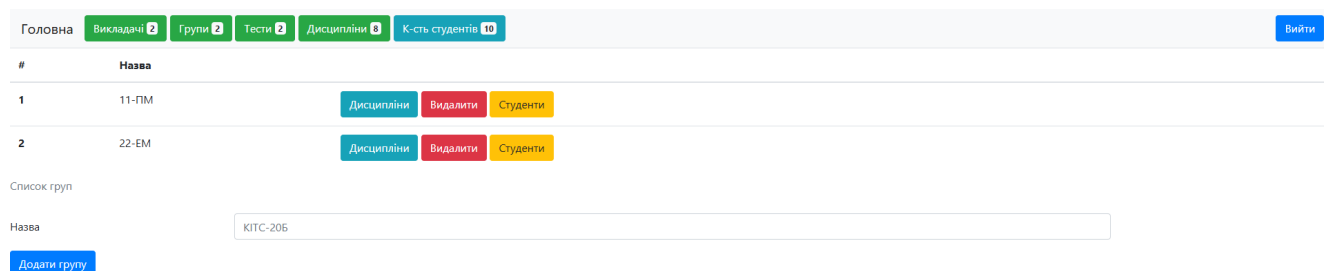


Рис. 3.4. Сторінка управління групами.

Тут він може позначити обрані групі дисципліни (рисунок 3.5.), створені користувачем «Викладач», видалити групу, додати студентів в групу (рисунок 3.6.), передивитись список вже існуючих студентів (рисунок 3.7.), задати тест групі (рисунок 3.8.) або подивитись таблицю вже заданих тестів (рисунок 3.9.).

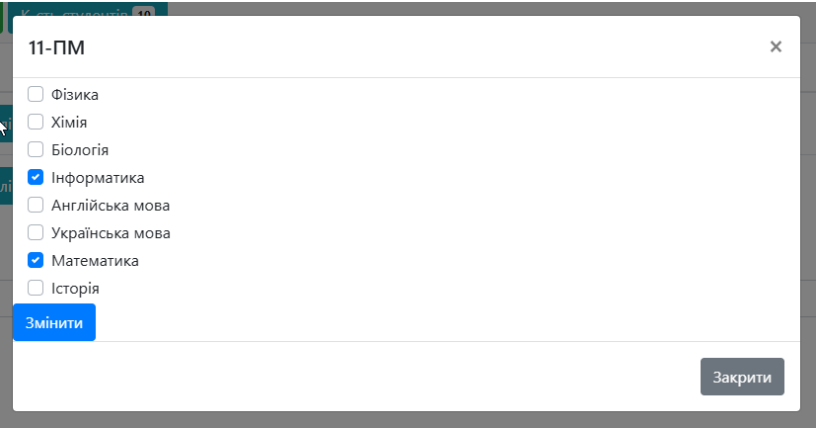


Рис. 3.5. Модальне вікно вибору дисциплін.

ПІБ	Іванов Іван Іванович
Група	11-ПМ
Логін	ivan2134
Пароль	1234567
Додати студента	

Рис. 3.6. Форма створення нового студента

#	ПІБ	Логін	Пароль	
1	Бабич Іван Васильович	PM-babich	123	Видалити
2	Бондар Денис Андрійович	PM-bondar	123	Видалити
3	Василенко Ігор Дмитрович	PM-vasilenko	123	Видалити
4	Горбань Михайло Маркович	PM-gorban	123	Видалити
5	Довбуш Михайло Петрович	PM-dovbush	123	Видалити

Рис. 3.7. Список студентів у групі

Задати тест

Оберіть тест

Математика 1

дд.мм.рррр --:--

Задати тест

Рис. 3.8. Форма задання тесту групі

Тести групи

ID	Назва	Здати До
7	Математика 1	21.06.2025 05:52:00
8	Інформатика 1	22.06.2025 05:52:00

Рис. 3.9. Таблиця заданих тестів групі

У списку студентів, користувач може обрати потрібного йому, та переглянути його оцінки з позначених йому дисциплін (рисунок 3.10.), та зберегти їх, натиснувши кнопку «Зберегти».

Студент Бабич Іван Васильович

Підсумкова1	Підсумкова2	Підсумкова3	Підсумкова4	Семестрова I	Підсумкова1	Підсумкова2	Підсумкова3	Підсумкова4	Підсумкова5	Семестрова II
12	11	12	9	11	8	9	11	7	10	9

Інформатика

Підсумкова1	Підсумкова2	Підсумкова3	Підсумкова4	Семестрова I	Підсумкова1	Підсумкова2	Підсумкова3	Підсумкова4	Підсумкова5	Семестрова II

Рис. 3.10. Список оцінок, обраного студента

Також адміністратор може додавати або видаляти дисципліни, з яких учні зможуть отримувати оцінки (рисунок 3.11.)

#	Назва дисципліни	
1	Фізика	Видалити
2	Хімія	Видалити
3	Біологія	Видалити

Рис. 3.11. Список дисциплін

Остання кнопка, якою може скористатись користувач «Адміністратор» - це кнопка «Вийти», у правій частині навігаційної панелі, яка деавторизує його, та виведе на сторінку авторизації.

3.2. КЕРІВНИЦТВО КОРИСТУВАЧА

Користувачі можуть авторизуватись як «Студенти» та «Викладачі». Авторизація «Викладача» починається з вікна авторизації (рисунок 3.1.). Після цього користувача перенаправляє на головну сторінку (рисунок 3.12.). Функції, якими може користуватись «Викладач», майже аналогічні функціям, якими користується «Адміністратор», за винятком створення нових викладачів. У нього також розташована навігаційна панель (рисунок 3.2.), у верхній частині екрану, але функції там інакші. Також, на відміну від адміністратора, викладач має можливість скористатись функцією зворотного зв'язку. Для цього потрібно заповнити форму зворотного зв'язку та натиснути кнопку «Відправити». Адміністратор отримує повідомлення на пошту та зв'язується з користувачем, у якого є якісь питання.

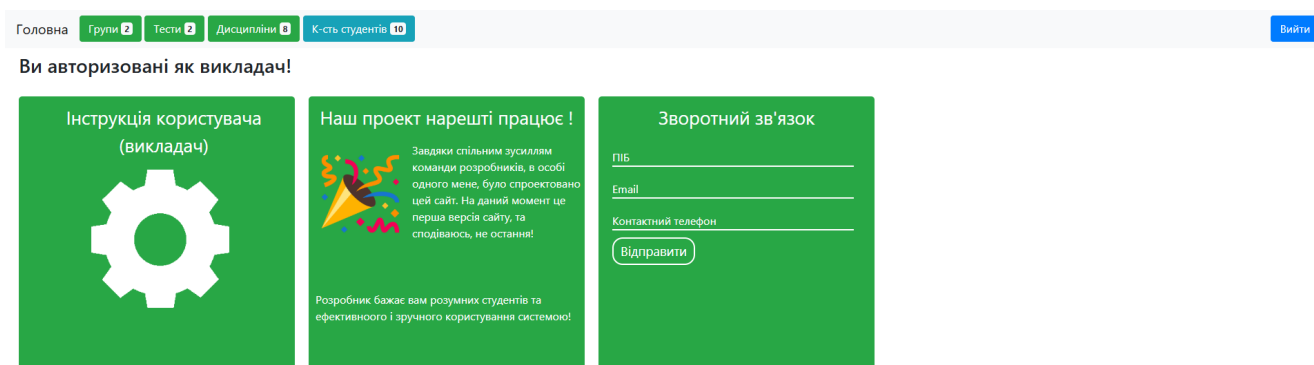


Рис.3.12. Головна сторінка користувача «Викладач»

Також, тут викладач може скористатись довідкою про користування програмою, натиснувши на блок «Інструкція користувача (викладач)». Після цього відкриється сторінка у новій вкладці з детальною інструкцією користувача (викладач) (рисунок 3.13.).

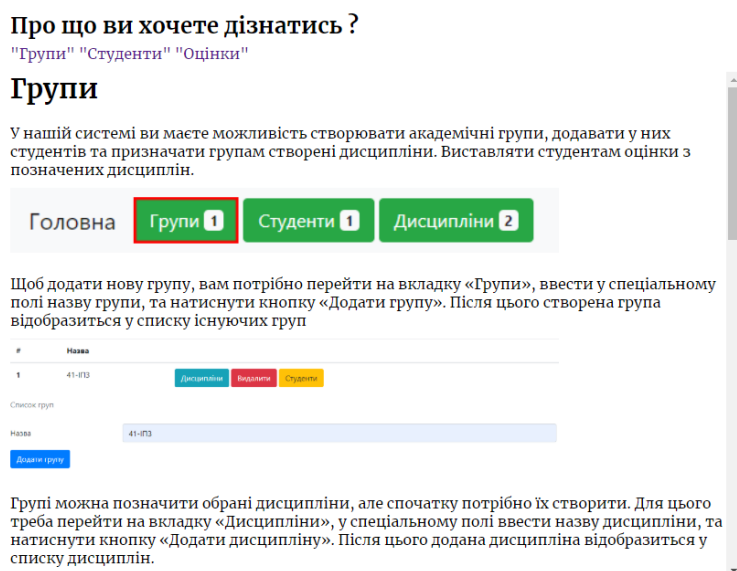


Рис. 3.13. Інструкція користувача (викладач)

Викладач може створювати групи, та наповнювати їх студентами, аналогічно адміністратору. Після вибору вкладки «Групи», користувач переходить на сторінку керування групами (рисунок 3.4.), де також може призначити групі обрані дисципліни.

Аналогічно попередньому типу користувача «Адміністратор», може створювати, переглядати та додавати студентів у групи.

Основною властивістю користувача типу «Вчитель» є створення тестів. Натискаючи на кнопку «Тести» можна перейти на сторінку перегляду вже створених тестів (рисунок 3.14., рисунок 3.15.). За допомогою кнопки «Створити» можна створити новий тест, за допомогою 4 типів питань: Звичайний, З декількома відповідями, Відповідність, З відкритою відповіддю. Можна додавати і видаляти питання, а також в питаннях додавати і видаляти варіанти відповіді (рисунок 3.16.).

Тести [Створити](#)

Список всіх тестів

ID	Назва тесту	
8	Інформатика 1	Переглянути
7	Математика 1	Переглянути

Рис. 3.14. Сторінка перегляду списку тестів

Тест: Інформатика 1

Що таке комп'ютер? **Бали: 1**

Правильні відповіді: Електронно-обчислювальна машина

☐ Монітор з великим розширенням
☐ Складна програма
☐ Електронно-обчислювальна машина
☐ Робот виконавець

Виберіть компоненти комп'ютера **Бали: 1**

Правильні відповіді: Материнка , Відеокарта , Клавіатура, Жорсткий диск

☐ Материнка
☐ Відеокарта
☐ Фотоапарат
☐ Клавіатура
☐ Жорсткий диск
☐ Планшет

Чи ви любите інформатику? **Бали: 5**

Відкрита відповідь.

Ваша відповідь

Рис. 4.15. Сторінка перегляду конкретного тесту

Створення нового тесту

Назва тесту

Тип: 1 — одна відповідь

Текст запитання

К-сть балів: 1

☐ Варіант 1
☐ Варіант 2
☐ Варіант 3
☐ Варіант 4

+

Тип: 2 — декілька відповідей

Текст запитання

К-сть балів: 1

☐ Варіант 1
☐ Варіант 2
☐ Варіант 3
☐ Варіант 4

+

Додати питання

Зберегти тест

Рис. 4.16. Сторінка створення тесту

Останнім типом користувача є «Студент», який створюється викладачем або адміністратором, та має обмежений функціонал. Після авторизації, на головній сторінці студенту одразу доступний журнал оцінок і список тестів доступних учню навігаційна панель з вкладками «Моя група», «Мої оцінки» (рисунок 3.17.) та за кнопкою пройти тест, можливо перейти на сторінку проходження тесту доступну тільки один раз (рисунок 3.18).

Головна

Моя група 5

Мої оцінки

Вийти

ПІБ	Логін	Пароль	Група
Бабич Іван Васильович	PM-babich	123	11-PM

Студент Бабич Іван Васильович

Підсумкова1	Підсумкова2	Підсумкова3	Підсумкова4	Семестрова I	Підсумкова1	Підсумкова2	Підсумкова3	Підсумкова4	Підсумкова5	Семестрова II

Інформатика

Підсумкова1	Підсумкова2	Підсумкова3	Підсумкова4	Семестрова I	Підсумкова1	Підсумкова2	Підсумкова3	Підсумкова4	Підсумкова5	Семестрова II

Математика

Тести

№	Назва	Набрані бали	Здати до	
8	Інформатика 1	2 (2025-06-17 05:53:41)	22.06.2025 05:52:00	Пройти тест
7	Математика 1		21.06.2025 05:52:00	Пройти тест

Рис. 3.17. Головна сторінка користувача «Студент».

Головна

Моя група 5

Мої оцінки

Вийти

Тест: Математика 1

Знак множення (1 бали)

☐ *

☐ /

☐ -

☐ +

Порахуйте (4 бали)

2+2

3*1

5-4

4/2

Напишіть відповідь (16/2+2)*3 (1 бали)

Ваша відповідь

Завершити тест

Рис. 3.18. Сторінка проходження тесту

При натисканні на вкладку «Моя група», користувач переходить на сторінку зі списком студентів в своїй групі (рисунок 3.19.).

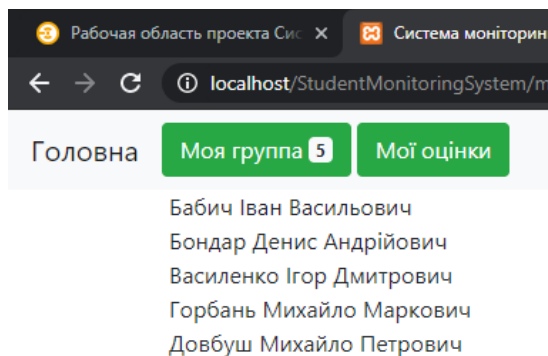


Рис. 3.19. Список студентів в групі.

3.3. КЕРІВНИЦТВО ПРОГРАМІСТА.

Для створення проєкту потрібно було виконати такі дії:

1. створити базу даних;
2. реалізувати логіку сайту;
3. реалізувати інтерфейсову частину сайту;
4. підключити базу даних до логічної частини;
5. наповнити базу даних.

Для підключення бази даних до логічної частини сайту, потрібно ввести дані про БД у змінні. У файлі *Base.php* (рисунок 3.20.).

```
$data = [  
    'host' => 'localhost',  
    'user' => '',  
    'pass' => '',  
    'base' => 'DB',  
    'charset' => 'utf8'  
];
```

Рис.3.20. Частина змісту файлу *Base.php*.

Файл *Base.php* підключається в усі інші файли, є основним та реалізовує основну логіку сайту. Такими чином, усі запити проходять через нього. Логіка сайту реалізована на мові *PHP* та *JavaScript*, з використанням бібліотеки *jQuery*. Елементи анімації, базова розмітка та інтерфейсу реалізована на мовах *HTML* та *CSS* з бібліотекою *Bootstrap*.

3.4. АНАЛІЗ РЕЗУЛЬТАТІВ РОБОТИ.

В процесі налагодження визначається якість втілення коду, зручність взаємодії з інтерфейсом, а також виявлення можливих вад у структурі сайту. Отримані результати надають можливість проаналізувати, наскільки добре інтерфейс відповідає сподіванням та вимогам кінцевого користувача.

Безпека даних забезпечується застосуванням атрибуту цілісності, який використовується для елементів, розроблених з використанням бібліотеки *jQuery*. Завдяки вбудованому хешу, цей механізм захищає код від несанкціонованих змін та гарантує його цілісність (див. рисунок 4.22).

```

<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
      integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T" crossorigin="anonymous">
<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
        integrity="sha384-q8i/X+965Dz00rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo" crossorigin="anonymous">
</script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.3/umd/popper.min.js"
        integrity="sha384-ZMP7rVo3mIykV+2+9J3UJ46jBk0WLaUAdn689aCwoqbBJiSnjAK/l8WvCWPim49" crossorigin="anonymous">
</script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.1.1/js/bootstrap.min.js"
        integrity="sha384-smHYKdLADwkXOn1EmN1qk/HfnUcbVRZyYmZ4qpPea6sjB/pTJ0euyQp0Mk8ck+5T" crossorigin="anonymous">
</script>
<script type="text/javascript" src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js"></script>

```

Рис. 3.21. Захисний атрибут integrity.

3.5. Висновки до розділу.

У даному розділі були розглянуті всі дії для усіх типів користувачів, які відвідують сайт. Було створено керівництво адміністратора, користувача та програміста, які облегшують розуміння роботи програмного продукту.

Висновки

В рамках цієї кваліфікаційної роботи успішно розроблено веб-платформу з метою моніторингу поточної успішності студентів у вищих навчальних закладах.

Основною метою дослідження було створення зручного, функціонального та адаптивного інструменту для освітнього тестування, який дозволяє ефективно відстежувати результати навчання в режимі реального часу.

У першому розділі розглянуто загальні теоретичні основи побудови автоматизованих систем у сфері освіти. Аналіз наукових джерел дозволив визначити актуальні підходи до розробки таких систем, висвітлити сучасні тенденції використання веб-технологій для реалізації навчальних платформ та з'ясувати ключові проблеми, що виникають у процесі цифровізації освітнього процесу.

Другий розділ зосереджений на детальному аналізі предметної області, визначенні функціональних вимог до системи та створенні логічної моделі. Було окреслено комплексну архітектуру майбутнього веб-застосунку, розглянуто основні модулі системи, спроектовано структуру бази даних та обґрунтовано вибір мов програмування, фреймворків та технологій для реалізації функціональності.

У третьому розділі було надано поглиблений огляд вибраних мов програмування (PHP, HTML, CSS, SQL, JavaScript з використанням jQuery) та фреймворку Bootstrap. Було представлено технічні характеристики використаних інструментів, описано переваги кожного компонента та обґрунтовано його роль у реалізації інтерактивного веб-інтерфейсу. Крім того, було визначено мінімальні вимоги до програмного та апаратного забезпечення, щоб користувачі могли без проблем використовувати систему незалежно від типу їхнього пристрою.

У четвертому розділі було представлено безпосередню реалізацію системи: було розглянуто принципи проектування інтерфейсу, реалізовано функції входу користувачів, перегляду результатів тестування, введення даних та відображення таблиці оцінок. Роботу платформи було проілюстровано за допомогою скріншотів,

а функціональні можливості системи описано у відповідь на запити користувачів. Особливу увагу було приділено питанням безпеки, зокрема цілісності коду, що забезпечується атрибутом integrity.

В результаті було створено ефективну веб-платформу для моніторингу академічної успішності, що дозволяє автоматизувати збір, зберігання та аналіз оцінок студентів. Розроблене рішення є зручним для користувача, інтуїтивно зрозумілим та безпечним. Кінцевий продукт має потенціал для масштабування, інтеграції з іншими освітніми системами та розширення функціональності в майбутньому.

Таким чином, поставлені цілі були повністю досягнуті, а результати кваліфікаційної роботи демонструють практичну цінність, інженерну доцільність та відповідність сучасним вимогам цифрового освітнього середовища.

ПЕРЕЛІК ПОСИЛАНЬ НА ДЖЕРЕЛА

1. Яскевич, Владислав Олександрович (2023) *JavaScriptбібліотека React Навчальний посібник для студентів спеціальності Комп'ютерні науки* Київський університет імені Бориса Грінченка, Україна, Київ.
<https://elibrary.kubg.edu.ua/id/eprint/48587/>
2. І.В Машкіна, ВО Абрамов, ТІ Носенко, ЄВ Іваніченко. Навчально-методичний посібник до виконання і захисту кваліфікаційної роботи бакалавра спеціальностей 122 Комп'ютерні науки для здобувачів вищої освіти денної форми навчання, Київський столичний університет імені Бориса Грінченка. Київ, 2024.- 43 с.
3. Теоретичні та практичні аспекти використання математичних методів та інформаційних технологій в освіті й науці: моногр. / за заг. ред. О. Литвин. — К.: Київ. ун-т ім. Б. Грінченка, 2021. — 332 с.
4. <https://www.dreamhost.com/blog/uk/vivchennia-php/> — **DreamHost Blog: Як швидко та безкоштовно вивчити PHP**
5. <https://xn----7sbbaqhlkm9ah9aiq.net/ua/news-new/jquery/> — **ABC Name: jQuery – використання і приклади**
6. <https://w3schoolsua.github.io/jquery/index.html> — **W3Schools UA: Підручник по jQuery**
7. <https://www.dreamhost.com/glossary/uk/veb-dyzayn-uk/jquery-uk/> — **DreamHost: Глосарій – jQuery**
8. <https://xn----7sbbaqhlkm9ah9aiq.net/ua/info/faq/shcho-take-jquery.html> — **FAQ jQuery – інформатика**
9. https://www.w3schools.com/jquery/jquery_intro.asp — **W3Schools: jQuery Introduction**
10. <https://bootstrap21.org/docs/5.0/forms/overview/> — **Bootstrap21.org: Огляд форм**
11. <https://byshep.com.ua/blogs/bootstrap-5-shcho-novogo> — **ByShep: Новини Bootstrap 5**

12. <https://nixsolutions.com/blog/js-skill-up/> — NIX Solutions: JS, HTML/CSS ресурси
13. <https://uh.ptv.ua/> — [Псевдо-ресурс placeholder для HTML/CSS довідника] *(треба замінити справжнім доступним)*
14. <https://itproger.com/course/html-css> — ITProger: Курс HTML/CSS українською
15. <https://uk.javascript.info/> — JavaScript.info: Сучасний підручник JS
16. <https://htmlacademy.ru/tutorial/php/mysql> — HTML Academy: PHP + MySQL