

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

ПОЯСНЮВАЛЬНА ЗАПИСКА

до кваліфікаційної роботи освітнього ступеня «бакалавр»
за спеціальністю 121 «Інженерія програмного забезпечення»
(освітня програма «Інженерія програмного забезпечення»)

на тему:

«Система з тестування учнів та управління навчальним закладом»

Виконав студент групи ІПЗ-19-3
МАРЦИНКОВСЬКИЙ Дмитро Русланович

Керівник роботи:
ПАНАРІНА Ірина Володимирівна

Консультант:
КУЗЬМЕНКО Олександр Вікторович

Рецензент:
КРАВЧЕНКО Світлана Миколаївна

Житомир – 2023

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

«ЗАТВЕРДЖУЮ»

В.о. зав. кафедри інженерії
програмного забезпечення
Андрій МОРОЗОВ
«14» лютого 2023 р.

ЗАВДАННЯ
на кваліфікаційну роботу

Здобувач вищої освіти: **МАРЦИНКОВСЬКИЙ Дмитро Русланович**

Керівник роботи: **ПАНАРІНА Ірина Володимирівна**

Тема роботи: **«Система з тестування учнів та управління навчальним закладом»,**

затверджена Наказом закладу вищої освіти від «14» лютого 2023 р. №70с

Вихідні дані для роботи: **використання мови програмування PHP, фреймворку Yii2, системи керування реляційними базами даних MySQL для створення системи спрощення і/або автоматизації тестування учнів та процесів в середині навчальних закладів.**

Консультанти з бакалаврської кваліфікаційної роботи із зазначенням розділів, що їх стосуються:

Розділ	Консультант	Завдання видав	Завдання прийняв
1	Панаріна І.В., Кузьменко О.В.	14.02.2023р.	14.02.2023р.
2	Панаріна І.В., Кузьменко О.В.	29.03.2023р.	29.03.2023р.
3	Панаріна І.В. , Кузьменко О.В.	25.05.2023р.	25.05.2023р.

Календарний план

№ з/п	Назва етапів випускної кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1	Постановка задачі	2 березня 2023 – 12 березня 2023	Виконано
2	Пошук, огляд та аналіз аналогічних розробок	12 березня 2023 – 19 березня 2023	Виконано
3	Формулювання технічного завдання та опрацювання літературних джерел	19 березня 2023 – 28 березня 2023	Виконано
4	Проектування структури системи	28 березня 2023 – 5 квітня 2023	Виконано
5	Написання програмного коду	5 квітня 2023 – 23 травня 2023	Виконано
6	Тестування роботи додатку	23 травня 2023 – 30 травня 2023	Виконано
7	Попередній захист	5 травня 2023	Виконано

РЕФЕРАТ

Випускна робота бакалавра представляє собою розроблений програмний продукт, а саме систему з тестування учнів та управління навчальним закладом та пояснювальної записки.

Пояснювальна записка до кваліфікаційної роботи складається з переліку умовних скорочень, вступу, трьох розділів, висновків, списку використаної літератури та додатку.

Текстова частина викладена на 50 сторінках друкованого тексту. Пояснювальна записка має 2 сторінки додатків. Список використаних джерел містить 20 найменувань і займає 2 сторінки. В роботі наведено 18 рисунків та 17 таблиць. Загальний обсяг роботи - 52 сторінок.

В роботі визначені головні завдання для розробки системи, проаналізовано аналогів продукту та методи вирішення поставленої задачі, обґрунтовано вибір архітектури системи. Представлено алгоритм роботи програмного комплексу, структуру та взаємодію його компонентів та описана програмна реалізація системи.

Ключові слова: PHP, YII2, MYSQL, СЕРВЕР, БД, PHPMYADMIN, CSS.

					ІПЗ.КР.Б – 121 – 23 – ПЗ			
Змн.	Арк.	№ докум.	Підпис	Дата	Система з тестування учнів та управління навчальним закладом	Літ.	Арк.	Аркушів
Розроб.		Марцинковський Д.Р						
Керівник		Панаріна І. В.					4	
Н. контр.		Тичина О. П.				ФІКТ Гр. ІПЗ-19-3[2]		
Зав. каф.		Морозов А. В.						

ABSTRACT

Bachelor's graduation work is a developed software product of a web platform for posting information products and an explanatory note.

The explanatory note to the diploma project consists of a list of conditional abbreviations, introduction, three sections, conclusions, a list of used literature and an appendix.

The text part is laid out on 50 pages of printed text. The explanatory note has 2 pages of appendices. The list of used sources contains 20 names and occupies pages. The paper contains 18 pictures and 17 tables. The total amount of work is 52 pages.

The main tasks for the development of the platform are defined in the work, analogues of the product and methods for solving the task are analyzed, the choice of the system architecture is justified. The algorithm of the software complex, the structure and interaction of its components are presented and the software implementation of the platform is described.

Keywords: PHP, Yii2, MySQL, SERVER, DATABASE, PHPMYADMIN, CSS.

					ІПЗ.КР.Б-121 – 23 - ПЗ	Арк.
						5
Змн.	Арк.	№ докум.	Підпис	Дата		

ЗМІСТ

РЕФЕРАТ	4
ВСТУП	7
РОЗДІЛ 1. АНАЛІЗ НАПРЯМКІВ ВИКОРИСТАННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ДЛЯ РЕАЛІЗАЦІЇ СИСТЕМИ ТЕСТУВАННЯ УЧНІВ ТА УПРАВЛІННЯ НАВЧАЛЬНИМ ЗАКЛАДОМ	8
1.1. Постановка задачі	8
1.2. Аналіз аналогів програмного продукту	9
1.3. Вибір архітектури веб-орієнтованої системи тестування учнів та управління навчальним закладом.	14
1.4. Обґрунтування вибору інструментальних засобів та вимоги до апаратного забезпечення.....	16
Висновки до розділу 1	16
РОЗДІЛ 2. ПРОЕКТУВАННЯ СИСТЕМИ ТЕСТУВАННЯ УЧНІВ ТА УПРАВЛІННЯ НАВЧАЛЬНИМ ЗАКЛАДОМ.	18
2.1. Визначення варіантів використання та об'єктно-орієнтованої структури системи	18
2.2. Розробка бази даних системи	21
2.3. Проектування та реалізація алгоритмів роботи системи.....	28
2.4. Реалізація системи з тестування учнів та управління навчальним закладом.....	33
Висновки до розділу 2.....	39
РОЗДІЛ 3. ІНТЕРФЕЙС ТА ПОРЯДОК РОБОТИ З СИСТЕМОЮ ТЕСТУВАННЯ УЧНІ ТА УПРАВЛІННЯ НАВЧАЛЬНИМ ЗАКЛАДОМ.....	40
3.1. Порядок встановлення та налаштування параметрів системи.....	40
3.2. Структура інтерфейсу системи тестування учнів та управління навчальним закладом.	41
3.3. Тестування роботи програмного комплексу.....	45
Висновки до розділу 3.....	46
ВИСНОВКИ	48
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	49
ДОДАТКИ	51

ВСТУП

Актуальність теми: Сьогодні все більше та більше факторів, змушує нас переносити нашу роботу, навчання та повсякденні справи до мережі Інтернет. Сфера освіти не стала виключенням. Проведення занять, тестування учнів, розміщення навчальних матеріалів та ведення бухгалтерського обліку також зазнали цифровізації. Оскільки на сьогодні діти та вчителі частіше знаходяться вдома, вчителям важче контролювати виконану роботу, спілкуватися та подавати новий матеріал. Тому розробка системи з тестування учнів та управління навчальним закладом є над актуальною темою.

Метою випускної кваліфікаційної роботи є закріплення та вдосконалення отриманих теоретичних та практичних знань з PHP, CSS, HTML, JS, SQL, які будуть використовуватися при розробці програмного продукту «Система з тестування учнів та управління навчальним закладом»

Результатом виконання випускної роботи є реалізоване програмне забезпечення, яке дозволить виконувати тестування учнів з певних предметів, ведення обліку, перегляду розкладу та статистичних даних. Для досягнення даної мети та результатів необхідно виконати низку поставлених задач:

- збір вимог та визначення конкретних функцій, можливостей та цілей;
- проектування системи, а саме дизайну, архітектури та інтерфейсу;
- проведення проектування бази даних;
- реалізація програмного продукту;
- впровадження заходів що до безпеки для захисту даних користувачів;
- проведення комплексного тестування та виправлення помилок;
- розгортання та обслуговування програмного продукту.

Предмет дослідження є методи та засоби проектування «Системи з тестування учнів та управління навчальним закладом».

Об'єктом дослідження є комплексна реалізація програмного продукту «Системи з тестування учнів та управління навчальним закладом».

					ІПЗ.КР.Б-121 – 23 - ПЗ	Арк.
						7
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 1. АНАЛІЗ НАПРЯМКІВ ВИКОРИСТАННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ДЛЯ РЕАЛІЗАЦІЇ СИСТЕМИ ТЕСТУВАННЯ УЧНІВ ТА УПРАВЛІННЯ НАВЧАЛЬНИМ ЗАКЛАДОМ

1.1. Постановка задачі

Метою створення системи тестування учнів та управління навчальним закладом є впорядкування та покращення різних аспектів навчального процесу. Це включає в себе аналіз та оцінювання, академічне планування та складання розкладу, відстеження успішності учнів, ведення документації та адміністративні завдання, аналіз даних та звітність. Система підвищує ефективність, точність та результативність, що призводить до покращення результатів навчання учнів та успішного управління навчальним закладом.

1. Розробити дизайн користувацького та адміністративного інтерфейсу.

1.1 Створити концепт програмного продукту;

1.2 Проектування та створення бази даних.;

1.3 Зверстати та застосувати елементи інтерфейсу:

1.3.1 Реєстрація та вхід;

1.3.2 Головна сторінка;

1.3.3 Профіль;

1.3.4 Мій клас;

1.3.5 Мій розклад;

1.3.6 Тестування;

1.3.7 Створення питань;

1.3.8 Перегляд статистики;

1.3.9 Класи;

1.3.10 Учні;

1.3.11 Вчителі;

1.3.12 Результат тестування;

2. Проектування та розробка основних контролерів:

2.1 Контролер сайту;

					ІПЗ.КР.Б-121 – 23 - ПЗ	Арк.
						8
Змн.	Арк.	№ докум.	Підпис	Дата		

- 2.2 Контролер класу;
- 2.3 Контролер новин;
- 2.4 Контролер питань;
- 2.5 Контролер розкладу;
- 2.6 Контролер учнів;
- 2.7 Контролер вчителів;
- 2.8 Контролер тестувань;
- 2.9 Контролер результатів;
- 3. Проектування та розробка основних модулів:
 - 3.1 Модуль учнів;
 - 3.2 Модуль вчителів;
 - 3.3 Модуль класів;
 - 3.4 Модуль предметів;
 - 3.5 Модуль тестів;
 - 3.6 Модуль розкладу;
 - 3.7 Модуль новин;
 - 3.8 Модуль реєстрації.

Результатом розробки даного програмного продукту є готова система для тестування учнів та управління навчальним закладом.

1.2. Аналіз аналогів програмного продукту

У цьому підрозділі ми проведемо аналіз освітніх платформ, які давно зарекомендували себе та мають велику популярність серед навчальних закладів. Дані сайти мають схожий задум, деякі спільні функції, але в той же час вони різні.

Вивчаючи дані платформи, основною метою є зрозуміти загальні особливості та характеристики, завдяки яким вони користуються попитом.

Для аналізу аналогів, було обрано три сайти, а саме:

Naurok.com.ua - це освітній портал, який пропонує різноманітні інструменти для навчання та тестування учнів, а також створення тестів, завдань, ресурсів; організація онлайн-тестувань; доступ до підручників і навчальних матеріалів.

					ІПЗ.КР.Б-121 – 23 - ПЗ	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

Vseosvita.ua - сайт надає інформацію і ресурси для вчителів та учнів українських шкіл, а також розміщення завдань для учнів, підручників, методичних матеріалів, новин та оголошень.

Eolymp.com - це онлайн-платформа для підготовки та проведення олімпіад з різних предметів з можливістю тестування за певними предметами і темами; рейтингова система; змагання та розв'язання завдань.

Дані платформи мають низку спільних переваг, але також і недоліків. Тож далі, ми розглянемо детальніше переваги та недоліки цих навчальних сайтів.

Naurok є найбільш відомим в Україні. Свою популярність набрав під час пандемії. Доступ до платформи - <https://naurok.ua/>

Особливості даного сайту:

- Можливість створювати тести та завдання: Naurok.com.ua дозволяє вчителям створювати тести, завдання та ресурси для учнів. Це дозволяє персоналізувати навчальний матеріал та адаптувати його до потреб учнів;
- Онлайн-тестування: Сайт надає можливість проводити онлайн-тестування, що дозволяє вчителям ефективно оцінювати знання та навички учнів;
- Доступ до підручників та навчальних матеріалів: Naurok.com.ua має багатофункціональну бібліотеку, де можна знайти підручники та навчальні матеріали з різних предметів;
- Спільнота вчителів: На сайті створена спільнота вчителів, де вчителі можуть ділитися досвідом, давати поради та співпрацювати один з одним.

Переваги:

- Багатофункціональність та різноманітність інструментів для тестування та навчання;
- Велика бібліотека підручників та навчальних матеріалів;
- Можливість онлайн-тестування;
- Спільнота вчителів, яка сприяє обміну досвідом та співпраці;

					ІПЗ.КР.Б-121 – 23 - ПЗ	Арк.
						10
Змн.	Арк.	№ докум.	Підпис	Дата		

Мінуси:

- Обмежені можливості безкоштовної версії, деякі додаткові функції можуть бути доступні лише за окрему плату.

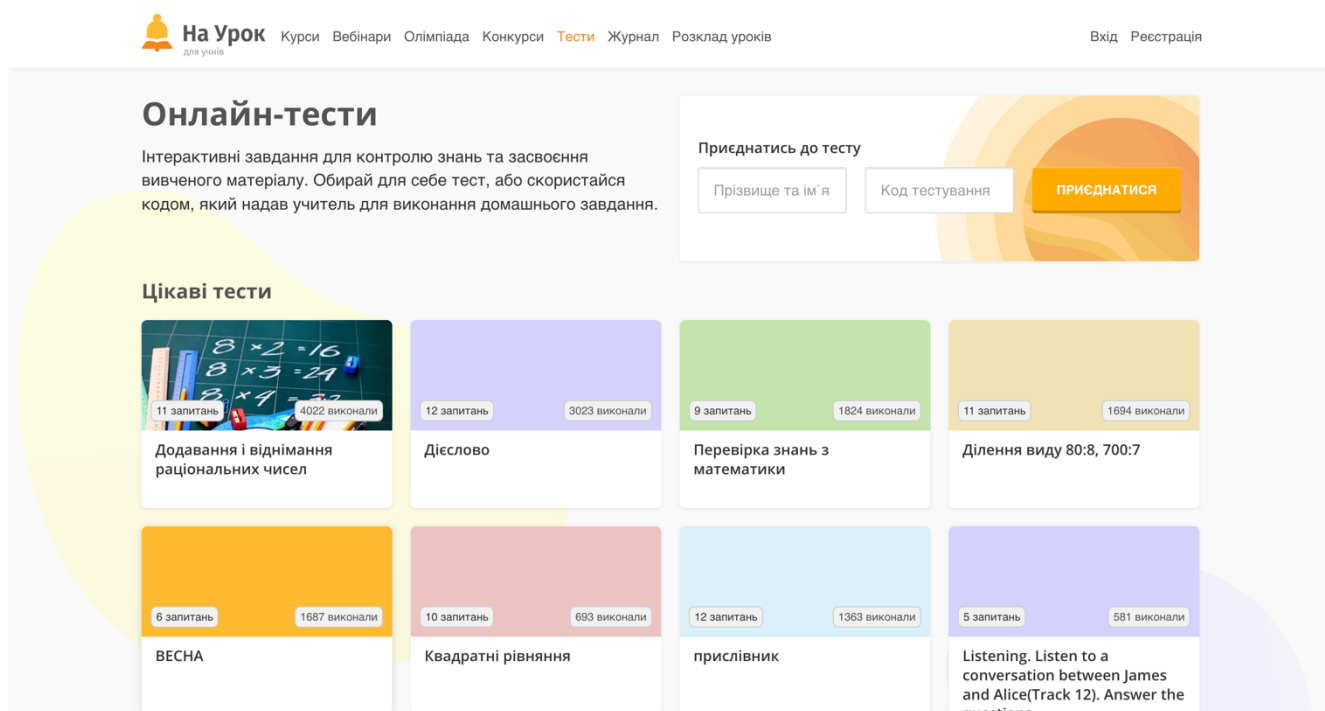


Рис. 1.1. Інтерфейс платформи На урок

Vseosvita.ua - український освітній веб-портал, який надає широкий спектр ресурсів та інструментів для вчителів, учнів та батьків для покращення навчання та освітнього процесу. Доступ до платформи - <https://vseosvita.ua/>

Переваги:

- На сайті розміщено багато освітніх матеріалів, таких як підручники, методичні посібники, рекомендації для вчителів та учнів;
- Vseosvita.ua надає інформацію про розклад уроків, дати іспитів та інші події в навчальних закладах;
- На сайті Vseosvita.ua є форуми та спільноти, де вчителі, учні та батьки можуть обмінюватися ідеями, досвідом та питаннями;
- Сайт регулярно оновлюється новинами та статтями з освітнього сектору.

Нижче наведено декілька недоліків даної платформи, які були знайдені під час її аналізу:

					ІПЗ.КР.Б-121 – 23 - ПЗ	Арк.
						11
Змн.	Арк.	№ докум.	Підпис	Дата		

- Обмежена доступність контенту: Деякі ресурси та функції можуть бути доступні лише після реєстрації або на платній основі;
- Брак інтерактивних інструментів;
- Потребує самостійної навігації.

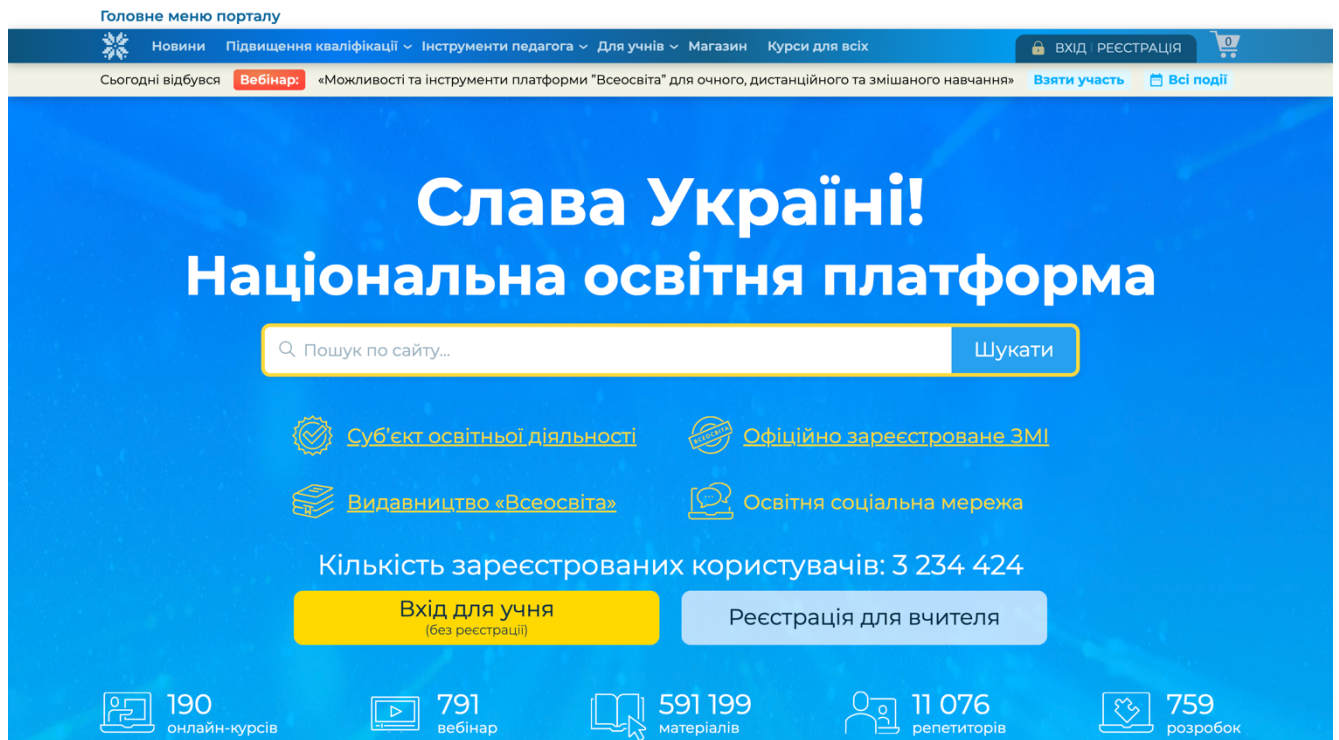


Рис. 1.2. Інтерфейс платформи Всеосвіта

Eolymr.com - онлайн-платформою для проведення та участі в змаганнях з інформатики, математики та інших наукових дисциплін. Доступ до платформи:

<https://www.eolymr.com/uk/>

Переваги:

- Сайт пропонує широкий вибір конкурсів та завдань з різних галузей, включаючи інформатику, математику, фізику, хімію тощо;
- Користувачі можуть проходити онлайн-тести на різні теми та порівнювати свої результати з іншими учасниками;
- Eolymr.com пропонує навчальні матеріали та пояснення до різних завдань і понять;
- Користувачі можуть навчатися на сайті в будь-який зручний для них час і у власному темпі;

Недоліки:

- Відсутність широкого спектру предметів: Сайт більше орієнтований на комп'ютерні науки та математику, і може бути обмежений в інших наукових дисциплінах;
- У порівнянні з деякими іншими освітніми платформами, eolymp.com може мати меншу спільноту користувачів і обмежені можливості для взаємодії та співпраці з іншими учасниками;
- Можлива складність завдань: Деякі завдання на сайті можуть бути високого рівня складності, що може стати проблемою для початківців або менш підготовлених учнів;

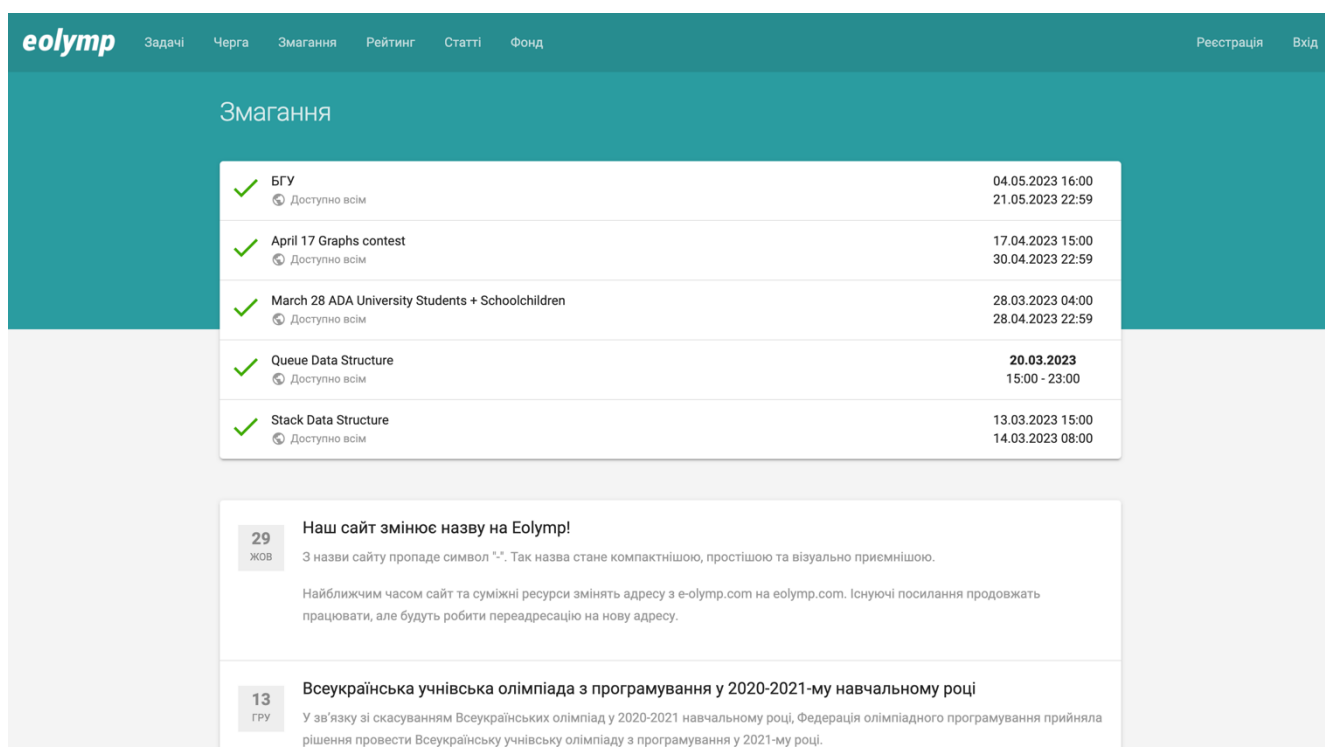


Рис. 1.3. Інтерфейс платформи Eolymp

Таблиця 1.1 Порівняння платформ

Платформа	Naurok	Vseosvita	Eolymp
Різноманітність тестів	+	+	+
Різноманітність освітніх матеріалів	+	+	-
Залучення батьків	-	+	-

Сертифікація	-	-	+
Доступність	+	+	+
Робота без мережі Інтернет	-	-	-
Чат спілкування з вчителем	-	+	-

Після проведення аналізу, можна ствердити, що кожен сайт має свої переваги та недоліки. А отже мною було сформовано список вимог до функціоналу мого сайту: створення, додавання, видалення та редагування вчителів, учнів, предметів та тестів. Проведення тестування учнів, з можливістю перегляду статистики. Створення та відображення розкладу занять. Для повноцінного функціонування потрібно реалізувати 3 ролі: учень, вчитель та адміністратор.

1.3. Вибір архітектури веб-орієнтованої системи тестування учнів та управління навчальним закладом.

Ще з самого початку розробки було зрозуміло, що ми будемо застосовувати архітектуру MVC (Model-View-Controller) для розмежування алгоритмічної частини від інтерфейсу та забезпечення спільного використання компонентів.

MVC є шаблоном проектування програмного забезпечення, який ділить додаток на три взаємопов'язані компоненти: Модель, Вид і Контролер. Модель представляє дані та бізнес-логіку програми.

У контексті системи навчального закладу Модель може обробляти такі завдання, як управління учнівськими записами, зберігання тестових даних та виконання розрахунків або аналізу даних.

Вид відповідає за представлення даних користувачам. Він має справу з інтерфейсом користувача і забезпечує інтерактивний і візуально привабливий спосіб взаємодії користувачів з системою. У моїй системі вид міг би включати сторінки для відображення інформації про учнів, результатів тестування та різних адміністративних функцій.

Контролер виступає посередником між Моделлю та Видом. Він отримує вхід користувача від View, обробляє його та відповідно оновлює Модель.

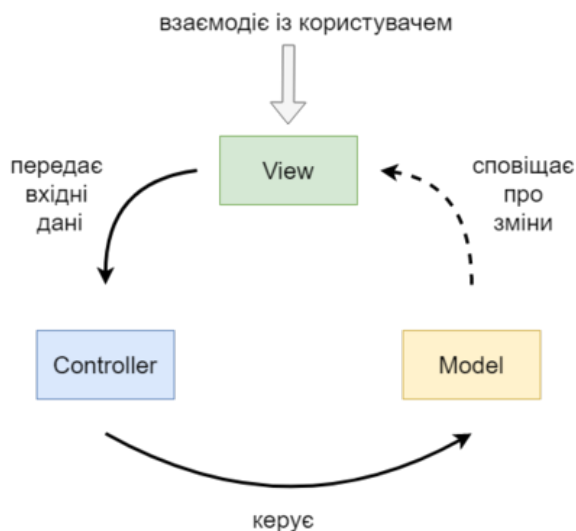


Рис. 1.4. Загальний принцип роботи MVC

У «Системі тестування учнів та управління навчальним закладом» шаблон MVC відігравав би вирішальну роль в організації коду та його розділення. Модель буде обробляти основні зберігання даних і маніпуляції, що дозволяє ефективно керувати інформацією учнів і тестувати дані.

Перегляд забезпечить зручні інтерфейси для учнів, вчителів та адміністраторів для взаємодії з системою, представляючи відповідну інформацію та параметри. Контролер буде обробляти логіку дій користувача, обробки запитів і оновлення моделі відповідно.

Прийнявши шаблон MVC, проект може досягти модульності та легкого розширення у майбутньому. Це дозволяє полегшити співпрацю між розробниками, так як різні компоненти можуть бути розроблені самостійно. Крім того, зміни або оновлення одного компонента, наприклад, View, не вимагають внесення змін до інших. В цілому, MVC забезпечує міцну основу для розробки масштабованої та надійної системи для тестування учнів та управління навчальним закладом.

1.4. Обґрунтування вибору інструментальних засобів та вимоги до апаратного забезпечення

Даний програмний комплекс повинен бути побудований на клієнт-серверній архітектурі з використанням веб-технологій, для яких не обов'язково мати ліцензування, або його проходити. Серверна частина буде розміщена на веб-сервері Open Server з системою керування базою даних phpMyAdmin з SQL [10]. Мовою програмування було обрано PHP із застосуванням фреймворку Yii2 [5].

Клієнтська частина будується на веб-технологіях, зокрема, використовується HTML, CSS і JavaScript [3]. Це дозволяє відтворювати інтерактивний інтерфейс у веб-браузері користувача.

Клієнтська складова програмного комплексу має бути повністю сумісною з різноманітними веб-браузерами, такими як Google Chrome, Safari та інші. Інтерфейс користувача повинен бути легким у використанні і привабливим, використовуючи останні дизайнерські техніки і підходи.

Важливо, щоб користувачу не довелося встановлювати різні додаткові програми, а було достатньо комп'ютера з підключенням до мережі, або телефону чи планшету. Також необхідно подбати про адаптацію до різних пристроїв.

Висновки до розділу 1

В ході розробки даного розділу було проведено аналіз вже існуючих систем для тестування учнів, проведення звітності чи управління навчальним закладом. Було визначено переваги та недоліки, що дало змогу визначити основні функції, які повинен мати мій проект. Адже визначення основних переваг аналогів, дозволить зібрати систему, що буде відповідати всім вимогам, а отже буде мати успіх серед користувачів.

Також було зроблено вибір архітектури веб-орієнтованої системи, а саме MVC (Model-View-Controller). Дана архітектура допоможе з легкістю розділити код для його подальшої організації. Крім цього MVC забезпечує міцну основу для розробки масштабованої та надійної системи, з фундаментом на майбутнє, адже проект можна буде з легкістю розширювати та залучати нових розробників.

					ІПЗ.КР.Б-121 – 23 – ПЗ	Арк.
						16
Змн.	Арк.	№ докум.	Підпис	Дата		

Не менш важливим було обґрунтування апаратних вимог, де ми вирішили, що розробка буде відбуватися на мові програмування PHP з використанням фреймворку Yii2. Клієнтська частина програми повинна бути максимально сумісною з різними веб-браузерами, такими як Mozilla Firefox та Google Chrome, для забезпечення зручного та надійного доступу до основних функцій. Під час розробки ми також ретельно враховуємо адаптивність до різних платформ, зокрема мобільних телефонів та планшетів, щоб забезпечити зручне використання програмного комплексу на будь-якому пристрої. Наша мета - забезпечити мобільність та зручність для користувачів, незалежно від пристрою, яким вони користуються.

Можна сказати, що ми визначили всі цілі, провели аналіз, обрали технологію, інструменти та архітектуру, що в результаті дало початок розвитку системи тестування учнів та управління навчальним закладом.

					ІПЗ.КР.Б-121 – 23 - ПЗ	Арк.
						17
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 2. ПРОЕКТУВАННЯ СИСТЕМИ ТЕСТУВАННЯ УЧНІВ ТА УПРАВЛІННЯ НАВЧАЛЬНИМ ЗАКЛАДОМ.

2.1. Визначення варіантів використання та об'єктно-орієнтованої структури системи

В даному розділі ми описуємо високорівневі вимоги, якими повинна володіти система тестування учнів та управління навчальним закладом, щоб мати можливість безперервно виконувати покладені на неї функції, тим самим відповідно задовольнивши всі зазначені стандарти, специфікації та інші формальні документи. Зазначені умови, яких повинен дотримуватись користувач для роботи з даною інформаційною системою та вказані основні ділові переваги даної ІС. Приведені можливості системи, обмеження та показники якості.

Бізнес вимоги

1. Основні цілі: проект створюється з метою спрощення процедури тестування учнів та ведення обліку учнів та вчителів з можливістю перегляду статистичних даних;
2. Представлення проекту: проект буде реалізовано у вигляді сайту, що містить структуровану інформацію по даній тематичній області.

Вимоги зовнішніх користувачів (Учень)

1. Можливість створення та перегляду власного профілю;
2. Можливість обирати предмет, тему та проходити тестування;
3. Можливість перегляду загальнодоступної інформації включаючи новини та важливі оголошення;
4. Можливість перегляду розкладу занять;
5. Можливість перегляду власних досягнень, після проходження тестів, у вигляді статистичних даних.

Вимоги зовнішніх користувачів (Вчитель)

1. Можливість створення та редагування власного профілю;
2. Можливість перегляду профілю;
3. Можливість створювати тестування, редагувати їх та видаляти;

					ІПЗ.КР.Б-121 – 23 - ПЗ	Арк.
						18
Змн.	Арк.	№ докум.	Підпис	Дата		

4. Можливість перегляду загальнодоступної інформації, новин та важливих оголошень;
5. Можливість створювати профіль учню, редагувати його та видаляти;
6. Можливість перегляду статистичних даних досягнень своїх учнів;
7. Можливість переглядати класи та їх розклад.

Вимоги зовнішніх користувачів (Адмін)

1. Можливість перегляду, редагування та видалення тестів;
2. Можливість перегляду, редагування, додавання та видалення загальнодоступної інформації новин та оголошень;
3. Можливість перегляду статистичних даних за різними тестами та опитуваннями;
4. Можливість створювати нові класи, предмети та розклад, а також відповідно редагувати та видаляти їх.

Характеристика об'єкта комп'ютеризації

Користувач додатку має достатньо можливостей та зручно розроблений інтерфейс для взаємодії вчителя з учнем та навпаки, а також адміністратором сайту.

Функціональні вимоги:

1. Авторизація в системі: Для користування всіма можливостями сайту, користувач потрібен пройти реєстрацію для присвоєння йому відповідної ролі (учень, вчитель, адміністратор). Є дві частини сайту для реєстрації, функціонал яких забезпечить правильне обрання ролі;
2. Можливість збереження інформації: Система повинна зберігати інформацію і надавати можливість керувати нею.

Нефункціональні вимоги:

1. Сприйняття.

					ІПЗ.КР.Б-121 – 23 - ПЗ	Арк.
						19
Змн.	Арк.	№ докум.	Підпис	Дата		

- Час, потрібний для навчання інструментами роботи з інформаційною системою для звичайних користувачів – 2-3 години, а для досвідчених – 1 година;
- Час відповіді системи для звичайних запитів не повинен перевищувати 5 секунд, а для більш складних запитів – 10 сек.;
- Інтерфейс представлення ІС повинен бути інтуїтивно зручним для користувача та не вимагати від нього додаткової підготовки.

2. Надійність

- Доступність – час, потрібний для обслуговування системи не повинен перевищувати 10% від загального часу роботи;
- Середній час безперервної роботи – 20 робочих днів;
- Максимальна норма помилок та дефектів в роботі системи – 1 помилка на 1000 запитів користувача;

3. Продуктивність

- Система повинна підтримувати мінімум 100 одночасно працюючих користувачів, пов'язаних з спільною базою даних;

4. Можливість експлуатації

- Масштабування – система повинна мати можливість збільшувати потужності (продуктивність), зі збільшенням користувачів таким чином, щоб це аж ні як негативно не відобразилося на її роботі.

Системні вимоги:

1. Вимоги до середовища виконання:

Система повинна задовольняти вимогам на комп'ютері, що знаходиться в наступній мінімальній комплектації:

- операційна система Windows 7 або вище, Linux 8.10 та вище, MacOS 10.6 та вище;
- 32-розрядний (x86) або 64-розрядний (x64) процесор із тактовою частотою 1 ГГц або вище;

					ІПЗ.КР.Б-121 – 23 - ПЗ	Арк.
						20
Змн.	Арк.	№ докум.	Підпис	Дата		

- 1 ГБ (для 32-розрядних процесорів) або 2 ГБ (для 64-розрядних процесорів) ОЗУ;
- 16 ГБ (для 32-розрядної системи) або 20 ГБ (для 64-розрядної системи) вільного місця на жорсткому диску;

2. Вимоги до СУБД та доступу до даних:

- У ядрі системи повинна бути представлена СУБД реляційного доступу.

Аналіз вимог користувачів дав підстави для визначення варіантів використання онлайн платформи для проходження тестів.

2.2. Розробка бази даних системи

Далі продовжимо розробку нашого додатку, розробляючи базу даних. Почнімо з проектування структури бази даних. Оскільки ми визначили задачі, які повинна виконувати наша база даних, можна сказати, що вона буде складатись з 16 таблиць.

А саме:

- auth_assignment;
- auth_item;
- auth_item_child;
- auth_rule;
- migration;
- user;
- teacher;
- student;
- schools;
- classrooms;
- subjects;
- schedule;
- news;
- exam_category;
- test_questions;

					ІПЗ.КР.Б-121 – 23 - ПЗ	Арк.
						21
Змн.	Арк.	№ докум.	Підпис	Дата		

- test_results.

Після створення всіх таблиць, що забезпечать повноцінну роботу нашого проекту, було спроектовано фізичну ERD діаграму.

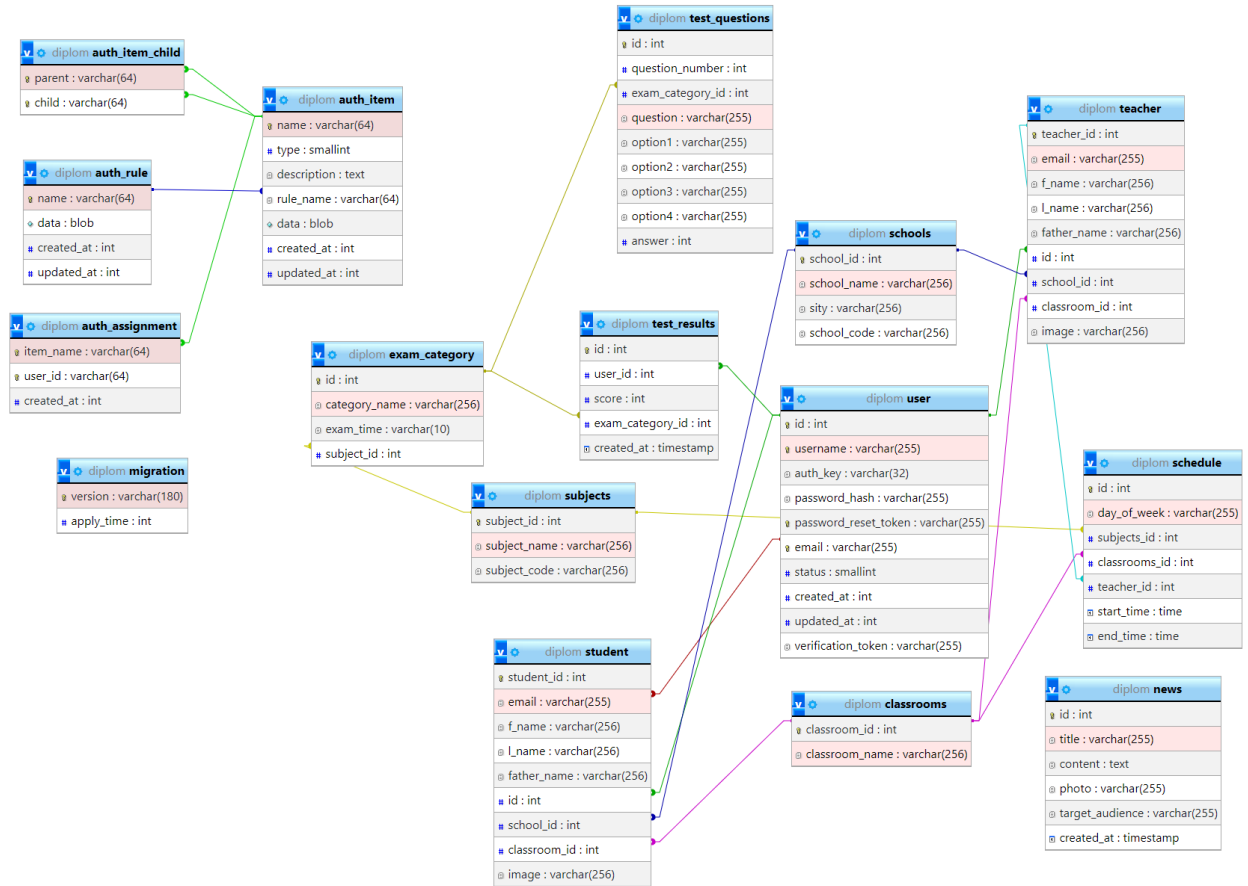


Рис. 2.1 Фізична ERD діаграма

Структура таблиць бази даних:

Таблиця 2.1

Таблиця auth_assignment

Поле	Ключ	Тип	Призначення
Item_name	PK	Varchar(64)	Назва ролі
User_id	FK	Varchar(64)	Порядковий номер користувача
Created_at		Int(11)	Дата присвоєння ролі користувачу

Таблиця 2.2

Таблиця auth_item

Поле	Ключ	Тип	Призначення
Name	RK	Varchar(64)	Назва ролі
Type		Smallint(6)	Тип. 1 – це роль, 2 – дозвіл
Description		text	Опис ролі чи дозволу
Rule_name	FK	Varchar(64)	Назва правила, яке застосовується до даного дозволу чи ролі
Data		blob	Додаткові дані в серіалізованому вигляді
Created_at		Int(11)	Дата створення
Updated_at		Int(11)	Дата оновлення запису

Таблиця 2.3

Таблиця auth_item_child

Поле	Ключ	Тип	Призначення
Parent	FK	Varchar(64)	Назва батьківського дозволу чи ролі
Child	FK	Varchar(64)	Назва дочірнього дозволу чи ролі

Таблиця 2.4

Таблиця auth_rule

Поле	Ключ	Тип	Призначення
name	RK	Varchar(64)	Назва правила
data		blob	Серіалізований об'єкт yii\rbac\Rule
created_at		Int(11)	Дата створення запису
updated_at		Int(11)	Дата оновлення запису

Таблиця 2.5

Таблиця user

Поле	Ключ	Тип	Призначення
id	РК	Int(11)	Код коритсувача
username		Varchar(255)	Логін
auth_key		Varchar(32)	Автентифікаційний код
password_hash		Varchar(255)	Захешований пароль
password_reset_token		Varchar(255)	Пароль для відновлення
email		Varchar(255)	Електронна пошта
status		Smallint(6)	Статус
created_at		Int(11)	Дата створення
updated_at		Int(11)	Дата оновлення
verification_token		Varchar(255)	Верифікаційний токен

Таблиця 2.6

Таблиця teacher

Поле	Ключ	Тип	Призначення
teacher_id	РК	Int(11)	Код
email		Varchar(255)	Пошта
f_name		Varchar(32)	Ім'я
l_name		Varchar(255)	Прізвище
father_name		Varchar(255)	По батькові
id	FK	Int(11)	Код користувача
school_id	FK	Int(11)	Код школи
classroom_id	FK	Int(11)	Код класу
image		Varchar(255)	Назва картинки

Таблиця 2.7

Таблиця student

Поле	Ключ	Тип	Призначення
student_id	РК	Int(11)	Код
email		Varchar(255)	Пошта
f_name		Varchar(32)	Ім'я
l_name		Varchar(255)	Прізвище
father_name		Varchar(255)	По батькові
id	FK	Int(11)	Код користувача
school_id	FK	Int(11)	Код школи
classroom_id	FK	Int(11)	Код класу
image		Varchar(255)	Назва картинки

Таблиця 2.8

Таблиця schools

Поле	Ключ	Тип	Призначення
schools_id	РК	Int(11)	Код
school_name		Varchar(255)	Назва школи
sity		Varchar(32)	Місто
school_code		Varchar(255)	Код, або скорочення

Таблиця 2.9

Таблиця classrooms

Поле	Ключ	Тип	Призначення
classroom_id	РК	Int(11)	Код
classroom_name		Varchar(255)	Назва класу

Таблиця 2.10

Таблиця subjects

Поле	Ключ	Тип	Призначення
subjects_id	РК	Int(11)	Код
subjects_name		Varchar(255)	Назва предмету
subjects_code		Varchar(32)	Код, або скорочення

Таблиця 2.11

Таблиця schedule

Поле	Ключ	Тип	Призначення
id	РК	Int(11)	Код
day_of_week		Varchar(255)	Номер дня тижня від 1 до 7.
subjects_id	FK	Int(11)	Код предмета
classrooms_id	FK	Int(11)	Код класу
teacher_id	FK	Int(11)	Код вчителя
start_time		time	Зазначення часу початку уроку
end_time		time	Зазначення часу кінця уроку

Таблиця 2.12

Таблиця news

Поле	Ключ	Тип	Призначення
id	РК	Int(11)	Код
title		Varchar(255)	Заголовок новини
content		text	Наповнення новини
photo		Varchar(255)	Посилання на фото
target_audience		Varchar(255)	Обрання аудиторії, яка побачить новину.
create_at		timestamp	Час збереження новини

Таблиця 2.13

Таблиця exam_category

Поле	Ключ	Тип	Призначення
id	PK	Int(11)	Код
category_name		Varchar(255)	Назва категорії
exam_time		Varchar(32)	Час який виділяється для тестування
subject_id	FK	Int(11)	Код предмета

Таблиця 2.14

Таблиця test_questions

Поле	Ключ	Тип	Призначення
id	PK	Int(11)	Код
question_number		Int(11)	Номер тесту для відображення
exam_category_id	FK	Int(11)	Код категорії
question		Varchar(255)	Назва питання
option1		Varchar(255)	Варіант відповіді 1
option2		Varchar(255)	Варіант відповіді 2
option3		Varchar(255)	Варіант відповіді 3
option4		Varchar(255)	Варіант відповіді 4
answer		Int(11)	Відповідь, яка відповідає числу від 1 до 4.

Таблиця 2.15

Таблиця test_results

Поле	Ключ	Тип	Призначення
id	PK	Int(11)	Код
user_id	FK	Int(11)	Код користувача

score		Int(11)	Результат тестування
exam_category_id	FK	Int(11)	Код категорії
created_at		timestamp	Час збереження відповіді

Таблиця 2.16

Таблиця migration

Поле	Ключ	Тип	Призначення
version		Varchar(180)	Актуальна версія міграції
apply_time		Int(11)	Час застосування

2.3. Проектування та реалізація алгоритмів роботи системи

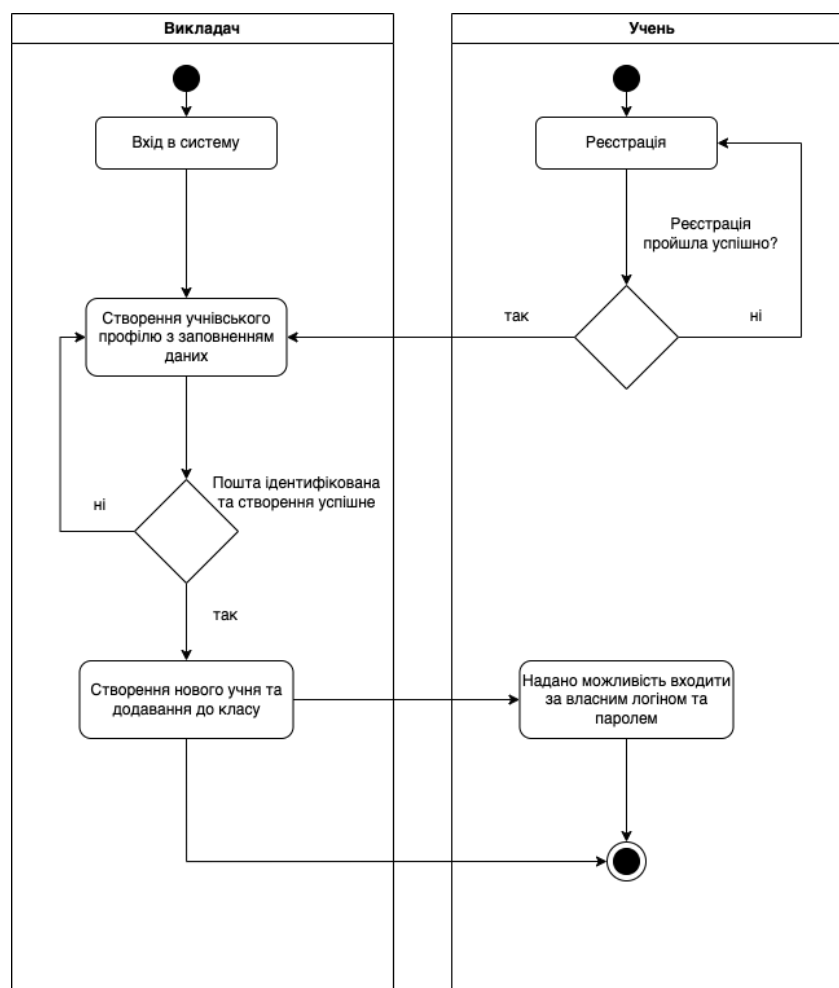


Рис. 2.2 Діаграма дій процесу реєстрації учня та створення йому учнівського профілю

Опис діаграми зображено на рисунку 2.3:

- Для користувачів з роллю Вчитель:
 - Реєстрація та вхід в систему;
 - Перегляд учнів, їх редагування та видалення;
 - Перегляд класів, їх редагування та видалення;
 - Перегляд інших вчителів, їх інформацію таку як пошта клас і так далі;
- Для користувачів з роллю Учень:
 - Вхід та реєстрація;
 - Перегляд власного профілю;
 - Перегляд вкладки Мій клас.

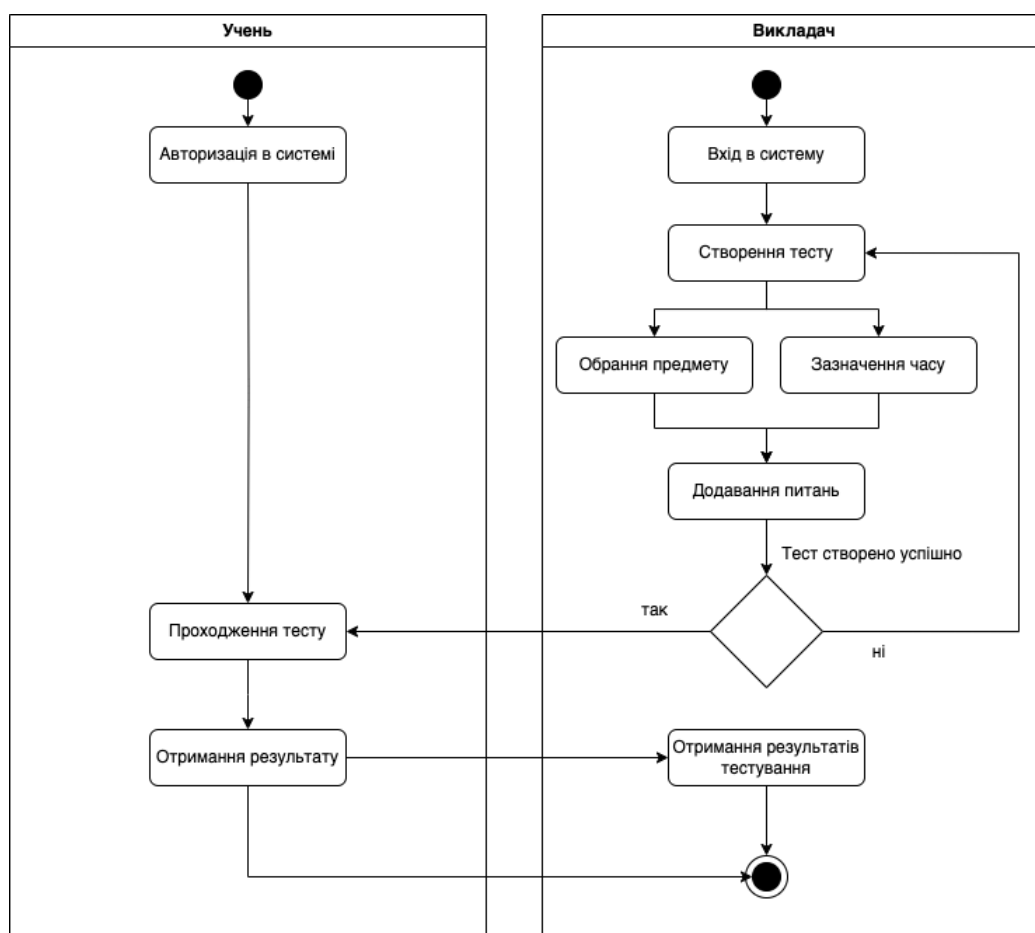


Рис. 2.3 Діаграма дій процесу створення тесту вчителем та його проходження учнем

Опис діаграми зображено на рисунку 2.4:

- Для користувачів з роллю Вчитель:
 - перегляд списку вже створених тестів з можливістю їх сортування за предметом, або пошуку по назві;
 - Створення тесту вказуючи назву, предмет та час;
 - Видалення та редагування вже створеного тесту;
 - Створення питання до тесту, де вказуються: назва, заповнюються варіанти відповідей, та обирається правильний варіант;
 - Видалення та редагування вже створених питань;
 - Перегляд статистики по вибраному учню, де вказується назва тесту та оцінка;
 - Перегляд загальної статистики розміщеної у вкладці Клас, де відображається середня оцінка, максимальна, мінімальна та кількість пройдених тестів;
- Для користувачів з роллю Учень:
 - перегляд списку доступних тестів з можливістю їх сортування за предметом, або пошуку по назві;
 - Проходження обраного тесту. В ході тестування відображається таймер, оновлення та закриття сторінки його не зупиняють;
 - Після проходження тесту, учню буде доступна сторінка з результатом та певними відомостями;
 - Якщо з головної обрати тест який вже пройдений, на сторінці буде відображатися результат;
 - Свої досягнення учень бачить на сторінці власного профілю.

					ІПЗ.КР.Б-121 – 23 - ПЗ	Арк.
						30
Змн.	Арк.	№ докум.	Підпис	Дата		

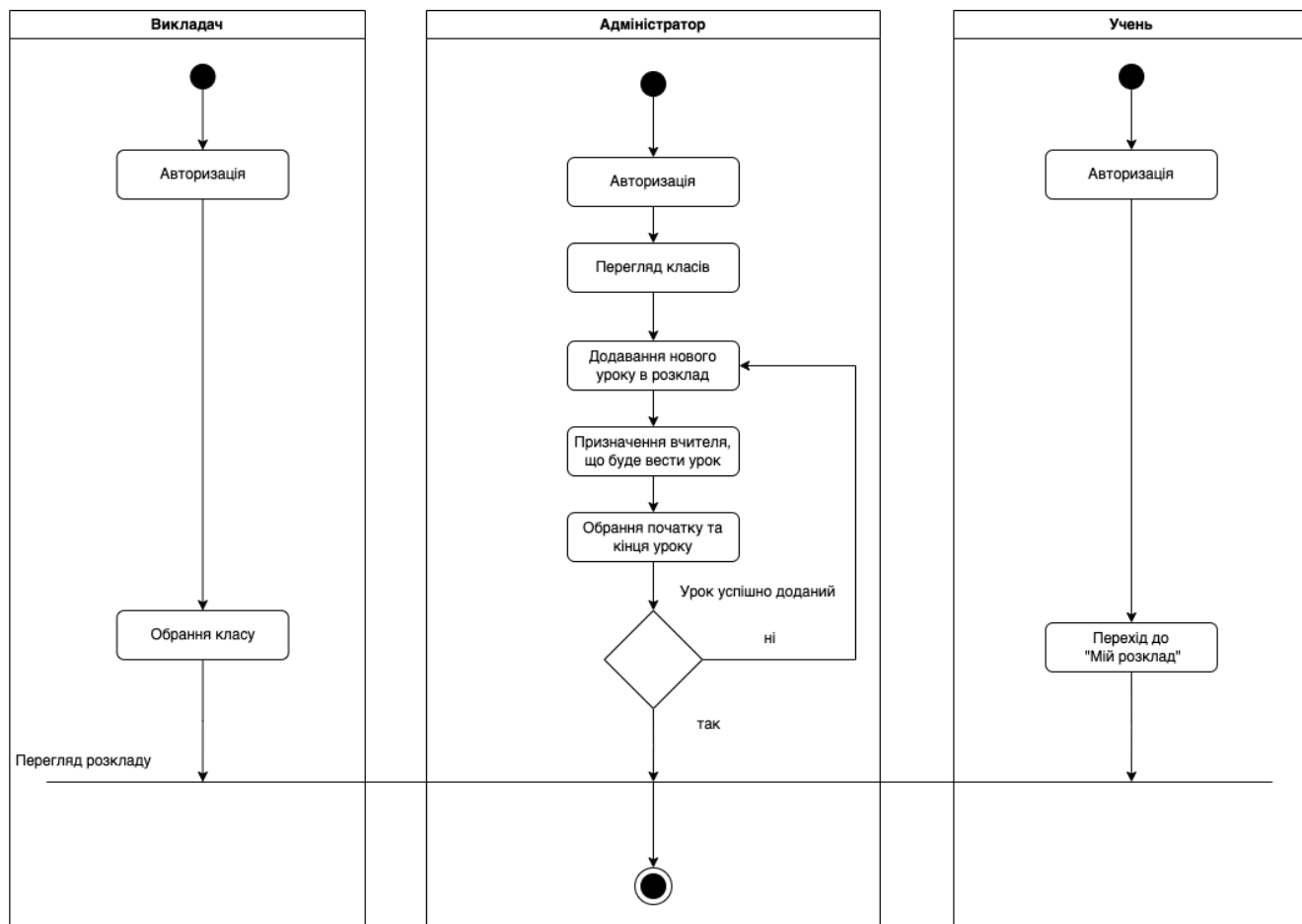


Рис. 2.4 Діаграма дій процесу створення тесту вчителем та його проходження учнем

Опис діаграми зображено на рисунку 2.5:

- Для користувачів з роллю Адміністратор:
 - Перегляд всіх класів та їх розкладів;
 - Можливість створювати нові класи за необхідністю;
 - Можливість створення нових предметів за необхідністю;
 - Обирати клас та додавати новий урок;
 - Призначати вчителів та час коли буде проведено урок;
 - Вносити зміни у вже створений розклад, редагуючи урок, або видаляючи його повністю;
 - Можливість видалення уроку, предмету, класу, вчителя та учня.
- Для користувачів з роллю Вчитель:
 - Обирати клас та переглядати його розклад;

- Для користувачів з роллю Учень:
 - Переглядати вкладку мій розклад, де відображається лише розклад класу в якому навчається учень.

Наступний та останній наш крок – це розробка діаграми послідовності системи. Діаграма послідовностей описує взаємодію між об'єктами або компонентами системи у певному порядку в часі. Вона показує послідовність повідомлень та дій, які вони виконують, і допомагає зрозуміти, як взаємодіють різні частини системи.

За допомогою діаграми послідовностей можна візуалізувати, як об'єкти обмінюються повідомленнями, які дії вони виконують та в якому порядку.

Для даного проекту було розроблено загальну діаграму послідовностей, яка представлена на рисунку 2.6.

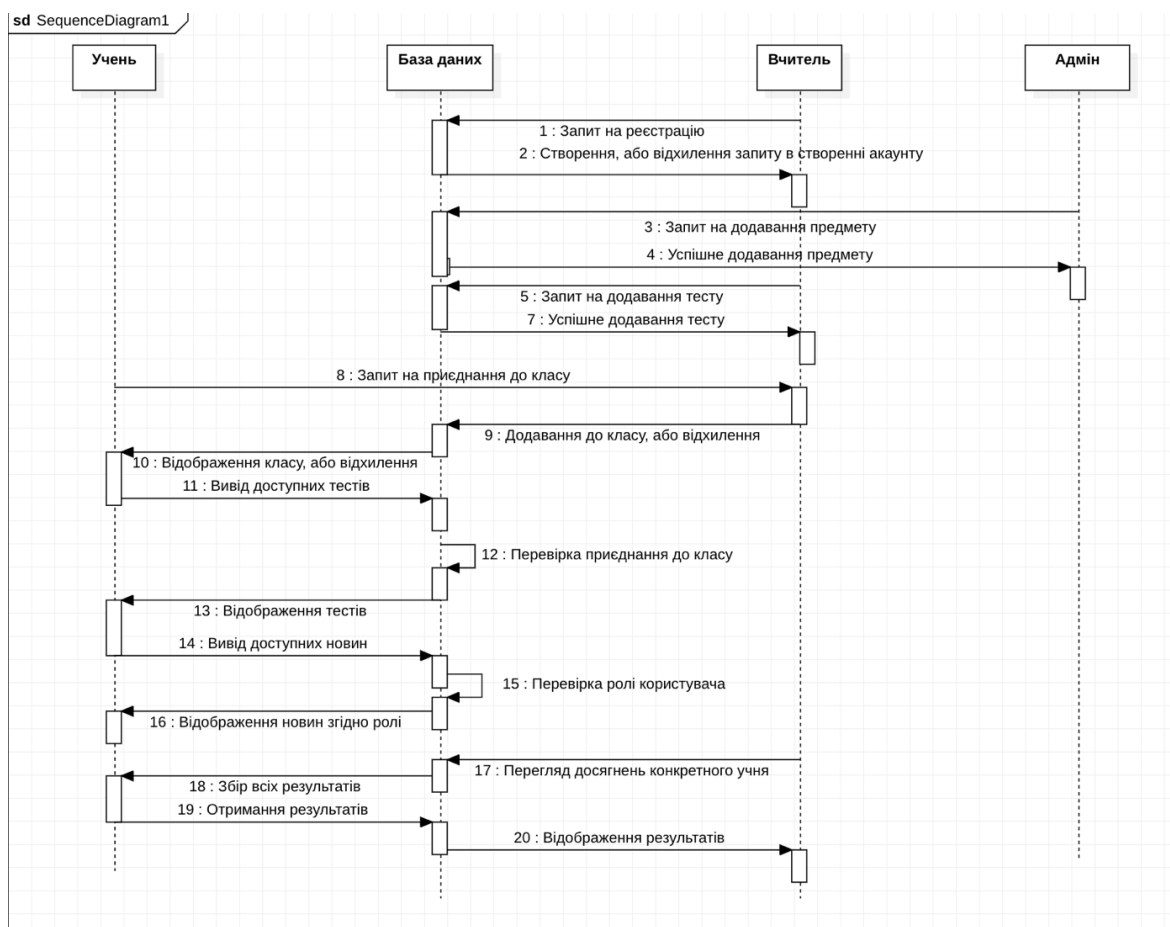


Рис. 2.5 Діаграма послідовностей системи.

2.4. Реалізація системи з тестування учнів та управління навчальним закладом

В даному розділі будуть розглянуті діаграми компонентів програмного забезпечення, які демонструють залежності та взаємозв'язки між різними елементами програмної системи. Ці компоненти можуть включати в себе вихідний код, бінарні компоненти та виконувані компоненти. Кожен модуль програмного забезпечення представлений окремою компонентою, яка відповідає за реалізацію класів, взаємозв'язків та функціональної поведінки системи.

Діаграми компонентів дозволяють візуалізувати структурні характеристики системи, але не відображають окремих екземплярів компонентів. Для цього можна використовувати діаграми розгортання, які показують, як компоненти фізично розміщуються та взаємодіють у реальному середовищі.

У цьому контексті, компоненти є конкретними фізичними або логічними частинами системи, що відповідають за реалізацію класів, зв'язків та функціональної поведінки програмного забезпечення. У зазначеному прикладі (див. Рисунок 2.7), компоненти включають:

- MySQL - база даних;
- Site Context - компонент, який відповідає за відображення контексту на веб-сторінці.;
- Student – тип користувача у системі (Учень);
- Admin – тип користувача у системі (Адміністратор);
- Teacher – тип користувача у системі (Вчитель);
- Models – компоненти, які є модулями в програмному коді:
 - Classrooms - модель для CRUD класів;
 - ClassroomsSearch – модель для класів;
 - Student - модель для CRUD учнів;
 - StudentSearch – модель для учнів;
 - Teacher - модель для CRUD вчителів;
 - TeacherSearch – модель для вчителів;

- Subjects - модель для CRUD предметів;
- TestResults - модель для CRUD результатів тестування;
- TestResultsSearch – модель для результатів тестування;
- TestQuestion - модель для CRUD питань;
- Schools - модель для CRUD школи;
- News - модель для CRUD новин;
- ExamCategory - модель для CRUD тестів;
- ExamCategorySearch – модель для тестів;
- Schedule - модель для CRUD розкладу;
- ScheduleSearch – модель для розкладу;
- User - модель для CRUD користувача;

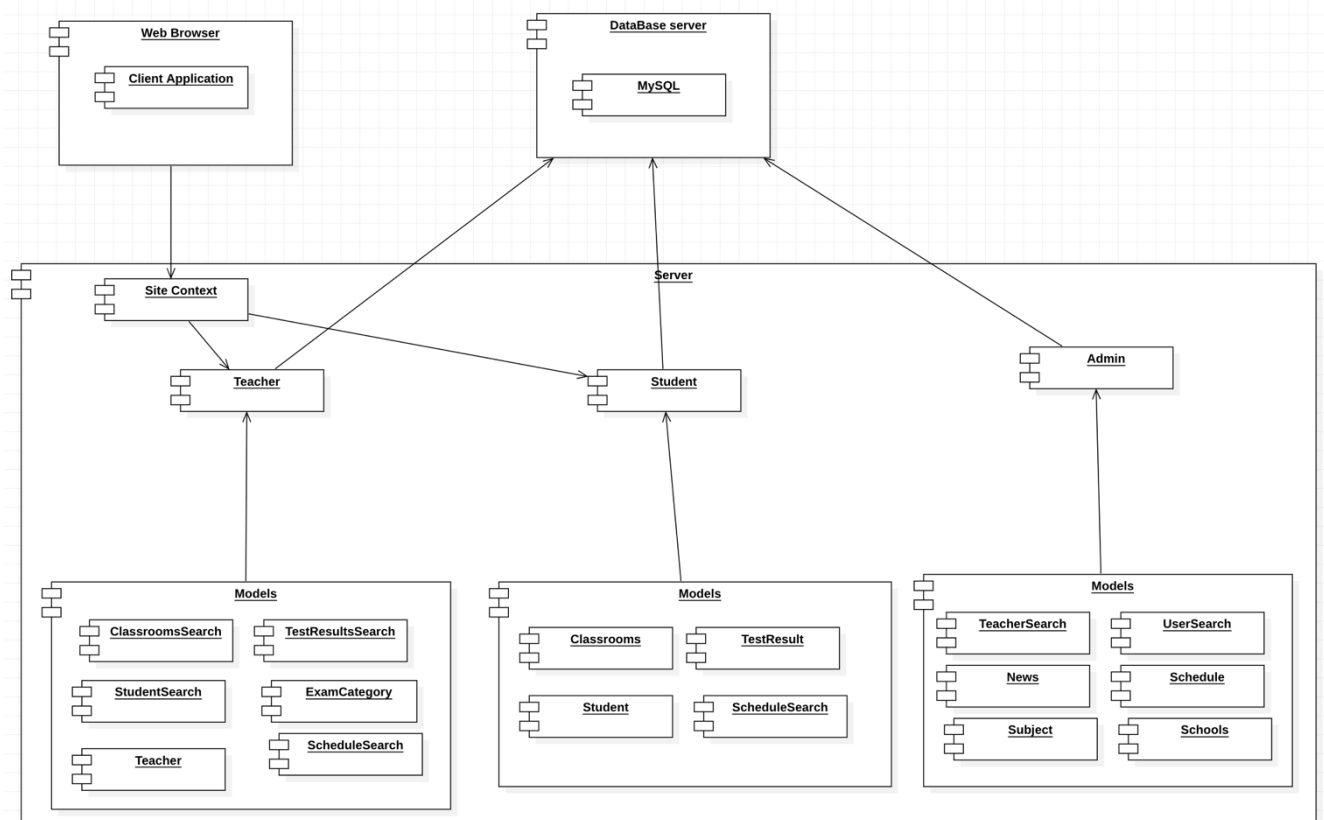


Рис. 2.6 Діаграма компонентів

Розпочнемо наше програмування. Для з'єднання нашої системи з базою даних нам необхідно зробити підключення яке відбувається у файлі config -> main-local.php [4].

```
return [
    'components' => [
        'db' => [
            'class' => 'yii\db\Connection',
            'dsn' => 'mysql:host=localhost;dbname=diplom',
            'username' => 'root',
            'password' => '',
            'charset' => 'utf8',
        ],
        'mailer' => [
            'class' => 'yii\swiftmailer\Mailer',
            'viewPath' => '@common/mail',
            'useFileTransport' => true,
        ],
    ],
];
```

Лістинг 3.1 - Метод підключення додатку до бази даних

Після підключення до бази даних, ми реалізовуємо реєстрацію та вхід окремо для учня та вчителя [18]. Після внесення даних виконується метод save(), далі дані записуються до таблиці. При реєстрації на відповідній стороні сайту, користувачу надається роль Вчитель, або Учень [11]. Далі наведено приклад надання ролі Вчитель:

```
public function signup()
{
    if ($this->validate()) {
        $user = new User();
        $user->username = $this->username;
        $user->email = $this->email;
        $user->setPassword($this->password);
        $user->generateAuthKey();
        $user->save(false);
        $auth = \Yii::$app->authManager;
        $authorRole = $auth->getRole('Teacher');
        $auth->assign($authorRole, $user->getId());
        return $user;
    }
    return null;
}
```

Лістинг 3.2 - Функція надання користувачу необхідної ролі

Далі було створено сторінки для відображення, створення, редагування та видалення учнів, вчителів, класів, питань та тестувань [8]. Відповідно до ролі користувача, кожен має різні права. Так наприклад адміністратор може додавати,

					ІПЗ.КР.Б-121 – 23 - ПЗ	Арк.
						35
Змн.	Арк.	№ докум.	Підпис	Дата		

редагувати та видаляти вчителів. Тоді як вчитель може лише переглядати дані своїх колег. В учня доступу до даних вчителів немає.

Наступний наш крок, це створення функціоналу для відображення тесту та результатів тестування [6]. Для цього було створено функцію `actionView`:

```
public function actionView($id)
{
    // Отримання поточного користувача і результатів тесту
    $userId = Yii::$app->user->id;
    $testResults = TestResults::find()
        ->where(['user_id' => $userId])
        ->andWhere(['exam_category_id' => $id])
        ->one();

    if ($testResults !== null) {
        // Користувач вже пройшов тест
        Yii::$app->session->setFlash('info', '');
        return $this->redirect(['result', 'id' => $testResults-
>exam_category_id]);
    } else {
        // Користувач ще не пройшов тест, відображаємо сторінку тестування
        $questions = TestQuestions::find()->where(['exam_category_id' => $id])-
>all();
        return $this->render('view', [
            'questions' => $questions,
            'testAlreadyTaken' => false,
        ]);
    }
}
```

Лістинг 3.3 - Функція відображення питань на сторінці тестування

Тепер ми можемо відображати питання та перевіряти чи користувач вже пройшов даний тест. Якщо тест пройдено, перенаправляємо користувача на сторінку `Result` з відповідним порядковим номером [15].

Далі створимо функцію для збереження відповідей учня та запис їх до бази даних. Для цього створюємо функцію `actionSubmit`:

```
public function actionSubmit()
{
    // Отримання поточного користувача і результатів тесту
    $userId = Yii::$app->user->id;
    $testResults = TestResults::find()->where(['user_id' => $userId,
'exam_category_id' => $id])->one();
    if ($testResults !== null) {
        // Користувач вже пройшов тест
        Yii::$app->session->setFlash('info', 'Тест вже пройдено');
    } else {
        // Користувач ще не пройшов тест
        $post = Yii::$app->request->post();
        $questions = TestQuestions::find()->where(['exam_category_id' =>
$post['TestQuestions'][0]['exam_category_id']])->all();
        $totalQuestions = count($questions);
        $correctAnswers = 0;
        if ($totalQuestions > 0) {
```

					ІПЗ.КР.Б-121 – 23 - ПЗ	Арк.
						36
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        $exam_category_id = $questions[0]->exam_category_id;
        foreach ($questions as $question) {
            $submittedAnswer = $post['TestQuestions'][$question-
>id]['answers'];
            if ($submittedAnswer == $question->answer) {
                $correctAnswers++;
            }
        }
        $score = round(($correctAnswers / $totalQuestions) * 100); //
Підрахунок оцінки в 100-бальному вигляді
        $testResults = new TestResults();
        $testResults->user_id = $userId;
        $testResults->score = $score; // Записуємо оцінку
        $testResults->exam_category_id = $exam_category_id;
        $testResults->created_at = date('Y-m-d H:i:s');
        ...
        return $this->redirect(['result', 'id' => $testResults->exam_category_id]);
    }

```

Лістинг 3.4 - Функція отримання відповідей та запису оцінки до бази даних

Тепер користувач після обрання відповідей може відправити форму, а його результати порахуються та запишуться в базу даних [14]. Користувача перенаправляє на сторінку результатів, яка реалізована за допомогою функції `actionResults`:

```

public function actionResult($id)
{
    // Отримайте оцінку для користувача із бази даних
    $userId = Yii::$app->user->id;
    $testResults = TestResults::find()
        ->where(['user_id' => $userId, 'exam_category_id' => $id])
        ->one();
    // Перевірка, чи знайдено результат тесту для користувача
    if ($testResults === null) {
        Yii::$app->session->setFlash('info', 'Екзанем не здано.');
```

Лістинг 3.5 - Функція перевірки наявності оцінки та відображення результату

Далі переходимо до реалізації розкладу. Оскільки у нас вже є моделі для вчителя, предмету, створюємо модель Розклад та пишемо функцію `actionCreate` для додавання уроку в розклад:

```

public function actionCreate($classroom_id)
{
    $model = new Schedule();
    $model->classrooms_id = $classroom_id; // Встановлюємо значення для
classrooms_id field
    if ($model->load(Yii::$app->request->post())) {
        // Перевіряємо чи викладач має вже урок в цей самий час
        $existingTeacherLesson = Schedule::find()
            ->andWhere(['teacher_id' => $model->teacher_id])
    }
}

```

```

->andWhere(['day_of_week' => $model->day_of_week])
->andWhere([
    'or',
    ['between', 'start_time', $model->start_time, $model->end_time],
    ['between', 'end_time', $model->start_time, $model->end_time],
])
->andWhere(['not', ['id' => $model->id]]) // виключаємо поточний урок
з перевірки, щоб не виникла помилка наявності
->one();
if ($existingTeacherLesson) {
    Yii::$app->session->setFlash('error', 'Помилка! Викладач вже має в цей
час урок! Поверніться та видаліть попередній.');
```

```

    return $this->refresh();
}
// Перевіряємо чи група має вже урок в цей самий час
$existingGroupLesson = Schedule::find()
->andWhere(['classrooms_id' => $model->classrooms_id])
->andWhere(['day_of_week' => $model->day_of_week])
->andWhere([
    'or',
    ['between', 'start_time', $model->start_time, $model->end_time],
    ['between', 'end_time', $model->start_time, $model->end_time],
])
->andWhere(['not', ['id' => $model->id]]) // виключаємо поточний урок
з перевірки, щоб не виникла помилка наявності
->one();
if ($existingGroupLesson) {
    Yii::$app->session->setFlash('error', 'Помилка! Група вже має в цей
час урок! Поверніться та видаліть попередній.');
```

```

    return $this->refresh();
}
if ($model->save()) {
    return $this->redirect(['index']);
}
}
return $this->render('create', [
    'model' => $model,
]);
}

```

Лістинг 3.6 – Метод для додавання уроку в розклад

Дана функція дозволяє зробити запис до бази даних нового уроку. Оскільки уроків безліч, важливим було створення перевірки, аби адміністратор не зміг поставити урок вчителю в день та час, коли той має урок. Даний метод працює завдяки запиту до MySQL [1].

Наступним реалізуємо функцію `actionIndex`. Оскільки в системі дві сторони, необхідно передбачити розміщення новин для учнів та вчителів окремо, адже не завжди учням потрібно бачити оголошення які призначені для вчителів. Для цього написано наступний код:

```

public function actionIndex()
{
    $userId = Yii::$app->user->id; // Отримуємо id поточного користувача

```

```

// Отримуємо інформацію з таблиці auth_assignment для поточного користувача
$authAssignments = AuthAssignment::find()->where(['user_id' => $userId])-
>all();
$userRoles = ArrayHelper::getColumn($authAssignments, 'item_name');
// Визначаємо цільову аудиторію на підставі ролей користувача
$targetAudience = [];
if (in_array('Student', $userRoles)) {
    $targetAudience[] = 'students';
}
if (in_array('Teacher', $userRoles)) {
    $targetAudience[] = 'staff';
}
// Відображаємо новини для цільової аудиторії
if (!empty($targetAudience)) {
    $news = News::find()->where(['target_audience' => $targetAudience])-
>all();
} else {
    $news = News::find()->all(); // Відображаємо всі новини, якщо не визначена
    цільова аудиторія
}
return $this->render('index', [
    'news' => $news,
]);
}

```

Лістинг 3.7 – Метод перевірки ролі користувача та відображення новин

Ми отримуємо порядковий номер поточного користувача, перевіряємо роль та відображаємо йому необхідні новини [2]. Додаванням, редагуванням та видаленням новин займається адміністратор. Для вчителів та учнів реалізовано лише перегляд.

Висновки до розділу 2

Дана система забезпечує функціонал для тестування учнів, збереження результатів тестування, ведення кадрових змін для вчителів та учнів, перегляд розкладу занять та новин і оголошень.

Система має дві частини, що забезпечує розподілення можливостей для учнів, та окремо для вчителів та адміністрації. Для певних функцій було створено декілька етапів перевірки, аби дані в базі не дублювалися та не викликали помилок.

Також було спроектовано алгоритми взаємодії між ролями, алгоритми дії процесів системи, а також було розроблено діаграму послідовності системи.

В останньому пункті було реалізовано весь запланований та спроектований функціонал в повноцінний працюючий програмний продукт. Представлено методи реалізації основних та важливих функцій системи.

					ІПЗ.КР.Б-121 – 23 - ПЗ	Арк.
						39
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 3. ІНТЕРФЕЙС ТА ПОРЯДОК РОБОТИ З СИСТЕМОЮ ТЕСТУВАННЯ УЧНІ ТА УПРАВЛІННЯ НАВЧАЛЬНИМ ЗАКЛАДОМ

3.1. Порядок встановлення та налаштування параметрів системи

Система для тестування учнів та управління навчальним закладом розміщена на сервері, а отже не потребує масштабного розгортання програми. Але в цей же час користувачу необхідно мати пристрій, який буде задовольняти мінімальним системним вимогам:

- операційна система Windows 7 або вище, Linux 8.10 та вище, MacOS 10.6 та вище;
- 32-розрядний (x86) або 64-розрядний (x64) процесор із тактовою частотою 1 ГГц або вище;
- 1 ГБ (для 32-розрядних процесорів) або 2 ГБ (для 64-розрядних процесорів) ОЗУ;
- 16 ГБ (для 32-розрядної системи) або 20 ГБ (для 64-розрядної системи) вільного місця на жорсткому диску;

На рисунку 3.1. представлена діаграма розгортання системи.

Діаграма розгортання – діаграма, на якій відображаються обчислювальні вузли під час роботи програми, компоненти, об'єкти, що виконуються на цих вузлах.

Діаграма розподілена на наступну структуру:

- User PC – веб-інтерфейс в браузері;
- DataBase Server – інтерфейс для взаємодії з базою даних;
- Web Server – веб-сервер, на якому працює програмне забезпечення;
- PHP – проєктований програмний продукт;

					ІПЗ.КР.Б-121 – 23 - ПЗ	Арк.
						40
Змн.	Арк.	№ докум.	Підпис	Дата		

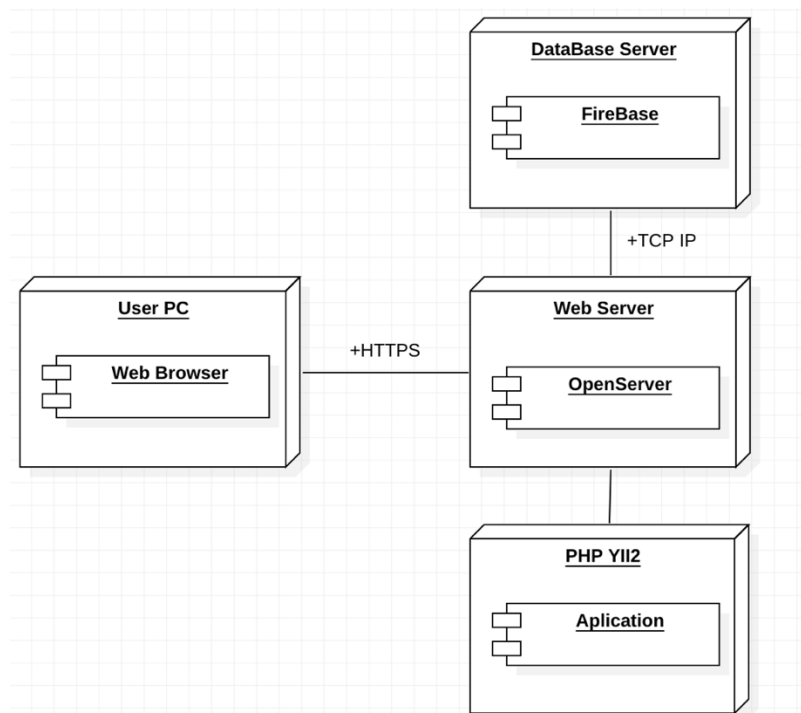


Рис. 3.1. Діаграма розгортання

3.2. Структура інтерфейсу системи тестування учнів та управління навчальним закладом.

Після входу на сайт, користувача як адміністратора, вчителя чи учня зустрічає головна сторінка. Де користувач може швидко зорієнтуватися та перейти на потрібну йому сторінку. Для створення дизайну використовуємо Bootstrap [12].

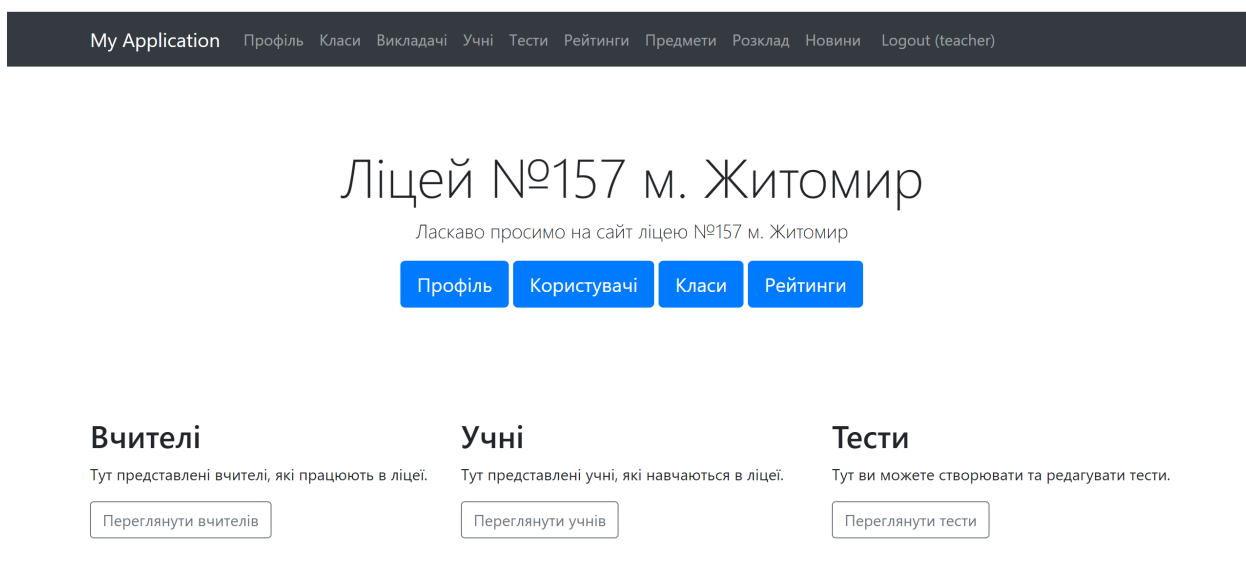


Рис. 3.2. Головна сторінка сайту після авторизації (версія для адміністратора та вчителя)

Кожному користувачу доступний власний кабінет. Для вчителів та адміністратора є можливість редагування власного профілю, для учня лише перегляд особистих даних [16].

My Application
Профіль
Класи
Викладачі
Учні
Тести
Рейтинги
Предмети
Розклад
Новини
Logout (teacher)

Головна /



Іваненко Олена

Редагувати

Логін	teacher
Прізвище	Олена
Ім'я	Іваненко
По батькові	Генадійна
Email	teacher@gmail.com
Клас	1-А
Школа	Житомирський ліцей №30

Рис. 3.3. Відображення профілю користувача (версія для адміністратора та вчителя)

Одною з основних функцій системи є відображення тестів з подальшою можливістю їх проходження. Для цього учню потрібно авторизуватися та перейти на вкладку Тестування.

My Application
Профіль
Мій клас
Тестування
Мій розклад
Новини
Logout (student)

Home / Тестування

Тестування

Оберіть один з доступних тестів:

Showing 1-5 of 5 items.

#	Назва	Час тестування	Предмет	
			Українська мова	
1	Формування правильної граматики: частини мови та їх функції	30	Українська мова	Розпочати
2	Розвиток лексичного запасу: слова-синоніми та антоніми	15	Українська мова	Розпочати
3	Вивчення правил орфографії: складні випадки правопису	25	Українська мова	Розпочати
4	Вправи на вдосконалення навичок читання та розуміння тексту	40	Українська мова	Розпочати

Рис. 3.4. Відображення списку доступних тестів, відсортованих за предметом

Учень може знайти тест за назвою, або зробити сортування за предметом. Після початку тесту, учень перейде на вкладку тестування. Запуститься таймер, який не можна збити оновивши вкладку, або вийшовши з тесту [19]. Дану функцію було реалізовано за допомогою скрипта JavaScript [9]. Час тестування зазначається вчителем при створенні тесту. Відповівши на всі питання, учень натискає кнопку Завершити. Далі він перейде на сторінку результатів, де побачить оцінку, час складання та іншу інформацію [13].

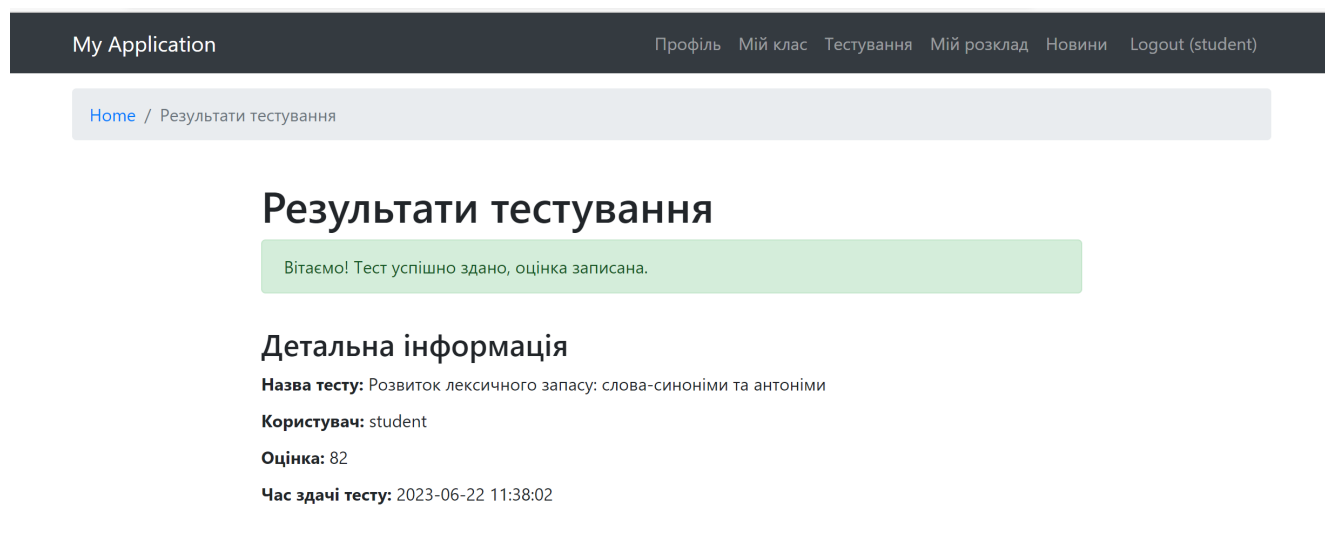


Рис. 3.5. Відображення оцінки учня, після завершення тестування

Для учня також доступний розділ Мій розклад. В ньому відображається актуальний розклад занять на тиждень. Він містить назву дисципліни, вчителя, кількість занять на тиждень та кількість годин.

Вчитель також може переглядати розклад, але для всіх класів. Для цього необхідно перейти на сторінку розкладу та обрати необхідний клас. Відображення розкладу для вчителя є аналогічним як для учня.

У адміністратора відкривається доступ до внесення змін, а саме редагування, видалення уроків, а також додавання нових класів та предметів. В разі, якщо вчитель вже має урок в цей час, адміністратор отримає помилки. Така ж помилка буде, якщо викладач буде додавати урок на час, який вже зайнятий. Для гарного та зрозумілого відображення розкладу, використовуємо Bootstrap v5.0 [7].

					ІПЗ.КР.Б-121 – 23 - ПЗ	Арк.
						43
Змн.	Арк.	№ докум.	Підпис	Дата		

My Application
Профіль
Класи
Викладачі
Учні
Тести
Рейтинги
Предмети
Розклад
Новини
Logout (admin)

1-A

Розклад уроків

Загальна кількість уроків: 12

Загальна кількість годин: 09:00

Час / День	Понеділок	Вівторок	Середа	Четвер	П'ятниця	Субота
08:30:00 - 09:15:00	Українська мова Олена І. Г.  	Українська мова Олена І. Г.  	Математика Олег П. Д.  	Українська мова Олена І. Г.  	Фізичне виховання Віталій К. С.  	
09:25:00 - 10:10:00	Математика Олег П. Д.  	Фізичне виховання Віталій К. С.  	Українська мова Олена І. Г.  	Математика Олег П. Д.  	Фізичне виховання Віталій К. С.  	
10:20:00 - 11:05:00	Іноземна мова (Англійська) Ольга Я. В.  		Фізичне виховання Віталій К. С.  			

Додати урок

Рис. 3.6. Відображення розкладу з можливістю його редагування для адміністратора

Також адміністратор, має можливість додавати новини та оголошення. Для цього йому потрібно перейти на спеціальну вкладку, де заповнюється заголовок, текст та вставляється посилання на фото. Також адміністратору необхідно обрати для кого буде відображатися дана новина, для учнів чи вчителів. Дата створення генерується автоматично. Для масштабування фото та стилізації тексту використовуємо CSS [17].

My Application
Профіль
Мій клас
Тестування
Мій розклад
Новини
Logout (student)

Новини



Реєстрація на літній табір відк...

Раді повідомити, що реєстрація на наш літній табір відкрита! Готуйтеся до захоплюючих пригод, нових

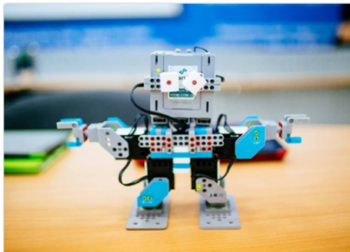
Переглянути



Відкриття футбольного сезону...

Увага, всі любителі футболу! Ми раді оголосити про відкриття нового футбольного сезону! Це час,

Переглянути



Майстер-клас з робототехніки...

Увага, учні! Ми раді оголосити про проведення захоплюючого майстер-класу з робототехніки! Якщо ви

Переглянути

Рис. 3.7. Відображення розкладу з можливістю його редагування для адміністратора

3.3. Тестування роботи програмного комплексу

Запуск програмного продукту не можливий без його тестування, тому наш наступний крок, саме перевірка функціоналу нашої системи. Це допоможе виявити помилки, перевірити функціонал та визначити загальну ефективність системи. В даному розділі ми розглянемо тест-кейси для основних функцій та методів системи, а також результат тестування загального функціоналу.

Спроба оновити сторінку тесту, або вийти з тесту задля оновлення таймера.

Передумови: учень переходить на сторінку «Тестування» та обирає тест. Відкривається нова сторінка, запускається відлік.

Кроки: спроба оновити сторінку, а також зайти та вийти з тесту.

Очікуваний результат: після оновлення сторінки таймер продовжує відлік не зупиняючись. Після виходу з тесту та повернення таймер продовжує відлік. Після закінчення часу, форма автоматично відправляється.

Фактичний результат: після оновлення сторінки таймер продовжує відлік не зупиняючись. Після виходу з тесту та повернення таймер продовжує відлік. Після закінчення часу, форма автоматично відправляється.

Додавання уроку в розклад, використовуючи вчителя, що вже має урок в цей час.

Передумови: адміністратор переходить на сторінку додавання нового уроку.

Кроки: по черзі заповнюються пункти – день тижня, предмет, вчитель, та час. Обираємо вчителя та час, що вже використовуються в розкладі.

Очікуваний результат: виникає сповіщення, про неможливість додати урок з даним вчителем на обраний час. Урок не додається, сторінка залишається з відкритою формою.

Фактичний результат: виникає сповіщення, про неможливість додати урок з даним вчителем на обраний час. Урок не додається, сторінка залишається з відкритою формою.

Створення нового учня, використовуючи пошту, що зареєстрована в системі.

Передумови: вчитель повинен перейти на сторінку «Учні» та натиснути «Додати учня».

Кроки: вчитель покроково заповнює ім'я, прізвище по батькові, обирає клас та заповнює пошту, яка вже зареєстрована в системі.

Очікуваний результат: профіль учня успішно додається. Увійшовши на сайт, учень переходить в профіль та бачить свої дані. Учень також приєднується до свого класу.

Фактичний результат: профіль учня успішно додається. Увійшовши на сайт, учень переходить в профіль та бачить свої дані. Учень також приєднується до свого класу.

Додавання оголошення для учня.

Передумови: адміністратор перебуває на сторінці створення новини.

Кроки: покроково заповнюємо поля – заголовок, зміст новини, додаємо посилання на зображення та обираємо аудиторію, для якої буде розміщено новину. В нашому випадку обирається для учнів.

Очікуваний результат: новина після публікації відображається на сторінці новини лише при авторизації з учнівського акаунту. Вчитель дану новину не бачить.

Фактичний результат: новина після публікації відображається на сторінці новини лише при авторизації з учнівського акаунту. Вчитель дану новину не бачить.

Висновки до розділу 3

В даному розділі було описано системні вимоги для комп'ютерної системи, щоб система для тестування учнів та управління навчальним закладом в повному обсязі виконувала свої функції. Також було спроектовано діаграму розгортання системи, на якій зображені компоненти та об'єкти, що використовуються під час роботи програми.

					ІПЗ.КР.Б-121 – 23 - ПЗ	Арк.
						46
Змн.	Арк.	№ докум.	Підпис	Дата		

Далі було продемонстровано інтерфейс та основні функції для кожної ролі. Також описано можливості системи. Завершальним етапом нашої розробки стало тестування. Було протестовано різні та важливі функції системи, які в результаті дали позитивний результат та розуміння, що програмний продукт повністю готовий до використання.

В результаті виконаних дій було отримано готовий продукт «Система з тестування учнів та управління навчальним закладом», який виконує весь свій функціонал, що зазначався раніше.

					ІПЗ.КР.Б-121 – 23 - ПЗ	Арк.
						47
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

В ході виконання випускної роботи бакалавра було спроектовано та реалізовано систему з тестування учнів та управління навчальним закладом. Під час виконання даної роботи було проведено детальний аналіз обраної тематики, а також порівняльну характеристику вже існуючих аналогів, або продуктів, що мають схожу ідею чи принцип роботи. В результаті чого було сформовано основні вимоги та визначено задачі для майбутньої системи.

Основним результатом дослідження є вибір архітектурного шаблону Model-View-Controller (MVC) для розробки системи. Цей вибір дозволяє ефективно розділити представлення даних, користувацький інтерфейс та логіку взаємодії, що забезпечує гнучкість, модульність та легкість у виправленні системи.

Було обґрунтовано рішення обрання саме таких інструментів розробки, а також описано апаратні вимоги для розробки системи. Для розміщення моєї системи на веб-сервері, було обрано Open Server та систему управління базами даних MySQL. Для написання коду системи було обрано мову програмування PHP та фреймворк Yii2. Для створення графічного інтерфейсу використовувались HTML, CSS, BOOTSTRAP5 та JavaScript.

В ході розробки системи було реалізовано тестування учнів з записом оцінки в базу даних. Можливість додавати та видаляти учнів та вчителів. Реалізовано 3 ролі, а саме для учні, вчителя та адміністратора. Реалізовано можливість перегляду статистики учнів, розкладу з елементами його редагування для адміністратора та сторінку новин.

Також були встановлені вимоги до комп'ютерної системи та розроблена діаграма розгортання системи. Було представлено інтерфейс системи, а також проведено тестування.

В результаті виконання бакалаврської роботи був розроблений програмний продукт «Система з тестування учнів та управління навчальним закладом», що відповідає всім вищезазначеним вимогам та функціоналу.

В результаті дана система може бути використана в майбутньому в якості основи для подальшого розвитку та вдосконалення системи.

					ІПЗ.КР.Б-121 – 23 - ПЗ	Арк.
						48
Змн.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. УЛІМАН Л. PHP and MySQL for Dynamic Web Sites: Visual QuickPro Guide. - Indianapolis: Peachpit Press, 2017. - 704 с.
2. Пауерс Д. PHP Solutions: Dynamic Web Design Made Easy. - Indianapolis: Friends of ED, 2014. - 528 с.
3. Склар Д. Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5. - Sebastopol: O'Reilly Media, 2014. - 812 с.
4. Сандерс Б. Yii2 by Example. - Birmingham: Packt Publishing, 2015. - 434 p.
5. Кочран Д. Yii2 Application Development Cookbook - Third Edition. - Birmingham: Packt Publishing, 2019. - 526 с.
6. Бутчер М. Bootstrap: Responsive Web Development. - Sebastopol: O'Reilly Media, 2013. - 250 с.
7. Летт Дж. Bootstrap 5: Building Responsive Web Pages and Applications with Bootstrap 5. - Birmingham: Packt Publishing, 2021. - 302 с.
8. Дакетт Дж. HTML & CSS: Design and Build Websites. - Indianapolis: John Wiley & Sons, 2011. - 512 с.
9. Фланаган Д. JavaScript: The Definitive Guide. - Sebastopol: O'Reilly Media, 2020. - 704 с.
10. Делісл М. Mastering phpMyAdmin 4.9: Effective administration and development of MySQL and MariaDB. - Birmingham: Packt Publishing, 2019. - 332 с.
11. Yii2 Documentation: Authorization [Електронний ресурс]/ yiiframework.com. – 2022-2023. – Режим доступу до ресурсу: <https://www.yiiframework.com/doc/guide/2.0/en/security-authorization>
12. Bootstrap Documentation: Components [Електронний ресурс]/ getbootstrap.com. – 2022-2023. – Режим доступу до ресурсу: <https://getbootstrap.com/docs/5.0/components/>
13. CSS-Tricks: CSS Tricks, Tips, and Techniques [Електронний ресурс] / css-tricks.com. – 2022-2023. – Режим доступу до ресурсу: <https://css-tricks.com/>

					ІПЗ.КР.Б-121 – 23 – ПЗ	Арк.
						49
Змн.	Арк.	№ докум.	Підпис	Дата		

14. phpMyAdmin Documentation: Official phpMyAdmin Documentation [Електронний ресурс] / docs.phpmyadmin.net. – 2022-2023. – Режим доступу до ресурсу: <https://docs.phpmyadmin.net/>
15. Yii2 RESTful API Guide [Електронний ресурс] / yiiframework.com. – 2022-2023. – Режим доступу до ресурсу: <https://www.yiiframework.com/doc/guide/2.0/en/rest-quick-start>
16. Bootstrap Documentation: Grid System [Електронний ресурс] / getbootstrap.com. – 2022-2023. – Режим доступу до ресурсу: <https://getbootstrap.com/docs/5.0/layout/grid/>
17. SS Tricks: CSS Grid Layout [Електронний ресурс] / css-tricks.com. – 2022-2023. – Режим доступу до ресурсу: <https://css-tricks.com/snippets/css/complete-guide-grid/>
18. MDN Web Docs: JavaScript Guide [Електронний ресурс] / developer.mozilla.org. – 2022-2023. – Режим доступу до ресурсу: <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide>
19. PHP Sessions: Everything You Need to Know [Електронний ресурс] / sitepoint.com. – 2022-2023. – Режим доступу до ресурсу: <https://www.sitepoint.com/php-sessions/>

ДОДАТКИ

					ІПЗ.КР.Б-121 – 23 - ПЗ	Арк.
						51
Змн.	Арк.	№ докум.	Підпис	Дата		

Варіанти використання додатку «Система тестування учнів та управління навчальним закладом»

