

Галицький фаховий коледж імені В'ячеслава Чорновола
відділення комп'ютерних технологій
циклова комісія інформатики та комп'ютерних дисциплін

ДОПУСТИТИ ДО ЗАХИСТУ

Завідувач відділенням

комп'ютерних технологій

Наталія СТЕФУРАК / _____ /
підпис

«___» _____ 2024 р.

ПОЯСНЮВАЛЬНА ЗАПИСКА

до кваліфікаційної роботи
освітньо-професійного ступеня «фаховий молодший бакалавр»
зі спеціальності 122 «Комп'ютерні науки»

на тему: «Розробка веб-системи підтримки навчального процесу»

Студент групи КН-41

Ярослав ГАВРИШКІВ _____
(підпис)

Керівник роботи

Наталія КУЛЬЧИНСЬКА _____
(підпис)

Консультанти:

з техніко-економічного
обґрунтування

Любов МЕЛЕНЧУК _____
(підпис)

нормоконтролер

Надія ГАВРИШКІВ _____
(підпис)

Тернопіль – 2024

Галицький фаховий коледж імені В'ячеслава Чорновола
відділення комп'ютерних технологій
циклова комісія інформатики та комп'ютерних дисциплін

ЗАТВЕРДЖУЮ

Завідувач відділенням

комп'ютерних технологій

Наталія СТЕФУРАК / _____ /
підпис
«__» _____ 2023 р.

ЗАВДАННЯ

на кваліфікаційну роботу

на здобуття освітньо-професійного ступеня «фаховий молодший бакалавр»
студенту Гавришківу Ярославу Васильовичу

(прізвище, ім'я та по-батькові студента)

1. Тема роботи Розробка веб-системи підтримки навчального процесу.

затверджено наказом по коледжу від “27” листопада 2023р., №234а-н

2. Термін здачі студентом завершеної роботи “__” _____ 2024р.

3. Вихідні дані до роботи: розробка веб-системи за допомогою веб-технологій HTML, CSS, JavaScript, PHP та MySQL

4. Перелік питань, які повинні бути розроблені:

а) основна частина: проєктування веб-системи, реалізація та тестування системи.

б) техніко-економічне обґрунтування: аналіз ринку, розрахунок витрат на проєктування, обґрунтування необхідності розробки.

5. Перелік графічного матеріалу: діаграми варіантів використання адміністратора, викладача, студента, макети інтерфейсу веб-системи.

6. Консультанти роботи: Любов Меленчук

Розділ	Консультанти	Підпис, дата	
		Завдання видано	Завдання прийнято
з техніко-економічного обґрунтування	<div></div> <div>(вчена ступінь, звання П.І.Б. консультанта)</div>		

КАЛЕНДАРНИЙ ПЛАН
виконання кваліфікаційної роботи

№ п/п	Найменування етапу	Терміни	
		початку	завершення
1.	Вибір теми та ознайомлення з вимогами кваліфікаційної роботи.	23.11.23	01.12.23
2.	Огляд типових рішень та написання відповідного розділу роботи.	05.12.23	10.01.24
3.	Дослідження технологій реалізації та написання відповідного розділу роботи.	28.02.24	20.04.24
4.	Розробка функціональних вимог до роботи та робота над структурою програмного продукту.	01.05.24	13.05.24
5.	Налаштування середовища реалізації та написання відповідного розділу роботи.	15.05.24	15.05.24
6.	Проектування програмного засобу (функціоналу, інтерфейсу).	15.05.24	27.05.24
7.	Тестування та налагодження веб-застосунку.	29.05.24	13.06.24
8.	Опрацювання техніко-економічного розділу роботи	14.06.24	14.06.24
9.	Оформлення пояснювальної записки	14.06.24	14.06.24
10.	Попередній захист кваліфікаційної роботи	17.06.24	17.06.24
11.	Підготовка до захисту кваліфікаційної роботи	17.06.24	24.06.24
12.	Захист кваліфікаційної роботи	26.06.24	26.06.24

7. Дата видачі “___” _____ 2023р. Керівник _____ /
Завдання прийняв до виконання Ярослав Гавришків.

Реферат

Кваліфікаційна робота. Розробка веб-системи підтримки навчального процесу. 60 сторінок, 42 рисунки, 2 додатки.

Об'єкт дослідження – веб-застосунок для підтримки навчання та його реалізація.

Метою роботи є розробка веб-додатку для підтримки навчання за допомогою веб-технологій HTML, CSS, JavaScript, PHP та MySQL.

Веб-застосунок повинен забезпечувати зручний доступ до інформації про академічні досягнення студентів, зокрема їхні оцінки за різними дисциплінами та інші показники успішності.

Крім того, необхідно спроектувати та реалізувати адміністративну частину додатку для ефективного управління оцінками студентів, користувачами та іншими аспектами академічного процесу.

Для досягнення поставленої мети використано різноманітні технології та інструменти веб-розробки.

Після завершення процесу розробки було отримано функціональний та готовий до використання веб-застосунок.

ВЕБ-ЗАСТОСУНОК, JOOMLA, БАЗА ДАНИХ, HTML, CSS, JAVASCRIPT, PHP, MYSQL.

Abstract

Qualification work. Development of a web system for supporting the educational process. Pages - 60, figures - 42, appendices - 2.

The object of research is a web application for educational support and its implementation.

The aim of the work is to develop a web application for educational support using web technologies HTML, CSS, JavaScript, PHP, and MySQL. The web application should provide convenient access to information about students' academic achievements, including their grades for various disciplines and other performance indicators. Additionally, it is necessary to design and implement an administrative part of the application for effective management of students' grades, users, and other aspects of the academic process.

To achieve the stated goal, various web development technologies and tools have been used.

Upon completion of the development process, a functional and ready-to-use web application was obtained.

WEB APPLICATION, JOOMLA, DATABASE, HTML, CSS,
JAVASCRIPT, PHP, MYSQL.

ЗМІСТ

Скорочення та умовні позначки	6
Вступ	7
1 Аналіз предметної області та постановка завдань	9
1.1 Опис предметної області	9
1.2 Аналіз наявних рішень	11
1.3 Аналіз вимог до програмного засобу та постановка завдання	14
2 Проєктування інформаційної системи	16
2.1 Проєктування функціонування інформаційної системи	16
2.2 Проєктування інтерфейсу	18
2.3 Проєктування адміністративної частини	20
2.4 Проєктування бази даних	22
3 Реалізація та тестування	26
3.1 Обґрунтування технологій і засобів реалізації	26
3.2 Реалізація інтерфейсу користувача	30
3.3 Реалізація функціоналу	36
3.4 Тестування	38
4 Техніко-економічне обґрунтування	47
4.1 Аналіз ринку	47
4.2 Розрахунок витрат на проєктування	50
4.3 Обґрунтування необхідності розробки	53
Висновки	56
Перелік джерел посилання	58
Додатки	59

					<i>КР. КН 24.552.04.000 ПЗ</i>			
Зм.	Арк.	№ докум.	Підпис	Дата				
Розроб.		ПІБ			Розробка веб-системи підтримки навчального процесу	Літ.	Арк.	Аркушів
Перев.		ПІБ					5	60
Рецензент.		ПІБ				ГФК.ВКТ.КН-41		
Н. Контр.		ПІБ						
Зав. від.		Стефурак Н.А.						

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

SQL – Structured Query Language

ІС – інформаційна система

СУБД – система управління базами даних

БД – база даних

ПК – персональний комп'ютер

ПЗ – програмне забезпечення

ІС – інформаційна система

CMS – система керування контентом

ПДВ – Податок на додану вартість

					<i>КР. КН 24.552.04.000 ПЗ</i>	Арк.
						6
Змн.	Арк.	№ докум.	Підпис	Дата		

ВСТУП

В нинішньому суспільстві Інтернет виконує дуже багато функцій і освітній процес не є винятком. Швидкими темпами розвивається дистанційна освіта, великого розмаїття набули різноманітні онлайн – курси. Варто відзначити, що, завдяки доступності підключення і його розповсюдженні, здійснювати навчальну діяльність можна у майже будь-якій точці світу.

Проте Інтернет не лише зумовив виникнення нових форм навчання, але здійснив вплив на уже існуючу систему освіти.

За допомогою всесвітньої мережі студенти можуть легко здійснювати обмін інформацією чи обговорення незрозумілих питань з іншими студентами чи викладачами. В той же час викладачі можуть контактувати з батьками чи опікунами підопічних, що забезпечує більший контроль на навчальним процесом.

Іншою властивістю мережі, яка відіграє значну роль в освіті є величезний обсяг інформації, яка знаходиться у вільному доступі. Для того щоб отримати відповідь на майже будь-яке питання достатньо знати адресу якогось із пошукових сайтів. І навіть якщо вам не вдалося щось знайти, завжди можна отримати кваліфіковану допомогу на профільних форумах.

Проте дуже часто Інтернет використовується лише як джерело інформації. Натомість хід освітнього процесу не відображається у мережі. Щоб дізнатися про свої оцінки здобувачам освіти доводиться набридати викладачам, проводити великий обсяг роботи із паперами. Галицький фаховий коледж в цьому питанні не є винятковим. Тому було вирішено створити систему підтримки навчального процесу, яка дозволить зберігати і відображати оцінки здобувачів освіти за допомогою веб-сайту навчального закладу.

Тема кваліфікаційної роботи – створення онлайн-платформи, яка буде сприяти покращенню процесу навчання, зокрема, забезпечувати зручний

					<i>КР. КН 24.552.04.000 ПЗ</i>	Арк.
						7
Змн.	Арк.	№ докум.	Підпис	Дата		

доступ до навчальних матеріалів, взаємодію між студентами та викладачами, а також зберігати та відображати інформацію про академічні досягнення.

Мета даної кваліфікаційної роботи полягає у розробці веб-базованої системи підтримки навчального процесу, яка дозволить зберігати та відображати оцінки студентів за допомогою веб-сайту навчального закладу. Це забезпечить автоматизацію процесу обліку оцінок, спростить доступ студентів до інформації про їхні результати, а також підвищить ефективність контролю з боку викладачів і адміністрації навчального закладу.

Об'єктом дослідження є процес впровадження веб-застосунку для підтримки навчання в навчальних закладах.

Предметом дослідження є структура, функціональні можливості та ефективність використання веб-застосунку для підтримки навчання в навчальних закладах.

Результати цієї кваліфікаційної роботи можуть бути застосовані в освітніх закладах різного рівня, таких як школи, коледжі, університети тощо. Веб-застосунок для підтримки навчання може бути використаний адміністрацією навчального закладу, викладачами та студентами для управління академічними даними, оцінювання здобувачів освіти, спрощення комунікації між учасниками навчального процесу та забезпечення доступу до навчальних матеріалів. Також, отримані результати можуть бути корисними для розробників програмного забезпечення, які працюють у сфері освіти, для подальшого вдосконалення та розширення функціоналу подібних систем.

					<i>КР. КН 24.552.04.000 ПЗ</i>	Арк.
						8
Змн.	Арк.	№ докум.	Підпис	Дата		

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАВДАНЬ

1.1 Опис предметної області

Галицький фаховий коледж імені В'ячеслава Чорновола – це навчальний заклад, який здійснює підготовку здобувачів освіти за різними спеціальностями. Навчальний процес передбачає вивчення низки дисциплін, за результатами яких студенти отримують оцінки.

Наразі процеси обліку й доступу до інформації про оцінки студентів не автоматизовані та не використовують сучасні можливості інформаційних технологій. Кінцеві оцінки з дисциплін вносяться лише у паперові залікові книжки здобувачів освіти та відомості успішності груп. Доступ студентів до відомостей успішності заборонений, а залікові книжки після закінчення семестру зберігаються у методистів відділень. Через це протягом значної частини навчального року студенти не мають змоги переглянути свою поточну успішність та історію оцінок.

Єдиним способом для здобувачів освіти дізнатися свої оцінки в такий період є звернення до куратора групи. Однак, ця процедура також є незручною та застарілою. Куратори мають збирати відомості успішності зі всіх дисциплін вручну, а студенти – звертатися до них особисто або телефоном, що створює додаткове навантаження для обох сторін.

Крім того, батьки студентів позбавлені можливості контролювати успішність своїх дітей дистанційно. Єдиний спосіб для них переглянути оцінки – це відвідування навчального закладу та особиста зустріч з куратором або викладачами.

Така ситуація є незадовільною з точки зору використання сучасних інформаційних технологій та створює значні незручності для студентів, батьків, викладачів та адміністративного персоналу коледжу. Тому назріла нагальна потреба в автоматизації процесів обліку успішності та забезпеченні

					<i>КР. КН 24.552.04.000 ПЗ</i>	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

зручного електронного доступу до цієї інформації для всіх зацікавлених сторін.

Такий стан речей є незручним не лише для студентів, а й для методистів відділень. Після завершення навчального року їм доводиться опрацьовувати велику кількість паперових документів, щоб сформувати звіт про успішність випускних груп.

Ситуація ускладнює також батьківський контроль за успішністю студентів. Батьки можуть переглянути оцінки своєї дитини лише безпосередньо в коледжі або за допомогою телефонного дзвінка, що не завжди є зручним способом.

Для вирішення цих проблем необхідно створити веб-застосунок, який дозволить автоматизувати процеси формування звітності та інформування студентів і батьків про успішність.

Впровадження такого додатку забезпечить централізоване зберігання всього масиву інформації про оцінки студентів у єдиній базі даних. Це надасть можливість маніпулювати даними за допомогою сучасних реляційних баз даних, зменшить обсяг рутинної роботи викладачів з внесення оцінок.

Веб-застосунок повинен надавати зручний інтерфейс для студентів та батьків щодо перегляду поточної успішності та історії оцінок. Викладачі матимуть можливість вносити та оновлювати оцінки в електронному форматі. Для методистів буде забезпечено автоматичне формування звітності про успішність випускних груп на основі даних з бази.

Таким чином, впровадження веб-додатку дозволить автоматизувати та модернізувати процеси управління навчальною успішністю в коледжі, підвищити їх ефективність та зручність для всіх учасників навчального процесу.

					<i>КР. КН 24.552.04.000 ПЗ</i>	Арк.
						10
Змн.	Арк.	№ докум.	Підпис	Дата		

1.2 Аналіз наявних рішень

Системи керування навчанням (СКН) або Learning Management Systems (LMS) – це веб-застосунки, призначені для створення, адміністрування та розповсюдження навчальних матеріалів в онлайн-форматі. Вони забезпечують спільний доступ до курсів, модулів, завдань, проектів та іншого навчального контенту. За допомогою СКН можна організовувати дистанційне навчання в режимі реального часу, проводячи онлайн-лекції, семінари, відстежуючи відвідуваність та успішність студентів.

Однією з найпоширеніших систем керування навчанням з відкритим вихідним кодом є Moodle (рис. 1.1). Це безкоштовна веб-платформа, розроблена на PHP з використанням баз даних SQL. Moodle надає викладачам інструменти для створення повноцінних сайтів для онлайн-освіти.



Рисунок 1.1 – Лого Moodle

Функціонал Moodle охоплює весь спектр можливостей, типових для електронних навчальних середовищ, зокрема прийом виконаних завдань, форуми для обговорень, завантаження файлів, систему оцінювання, обмін повідомленнями, календар подій, онлайн-тестування тощо.

Розробники можуть розширювати функціональність Moodle, створюючи нові модулі різних типів:

- діяльності;

					<i>КР. КН 24.552.04.000 ПЗ</i>	Арк.
						11
Змн.	Арк.	№ докум.	Підпис	Дата		

- ресурси;
- питання для тестів;
- поля для баз даних;
- теми оформлення;
- методи автентифікації та зарахування на курси;
- фільтри для контенту.

Така модульна архітектура робить Moodle гнучкою системою, яку можна налаштовувати відповідно до специфічних потреб навчального процесу.

Іншою популярною веб-орієнтованою системою керування навчанням є ATutor (рис. 1.2). Вона розробляється та підтримується з 2001 року Грегом Геєм, Джоелом Кроненбергом та Гайді Гейзелтон з Університету Торонто. ATutor поширюється на умовах вільної ліцензії GNU GPL, що дозволяє вільно використовувати, змінювати та доповнювати її код.



Рисунок 1.2 – Лого Atutor

У ATutor визначено 3 типи користувачів:

- студенти;
- викладачі;

					<i>КР. КН 24.552.04.000 ПЗ</i>	Арк.
						12
Змн.	Арк.	№ докум.	Підпис	Дата		

– адміністратори.

Для кожної категорії передбачений різний функціонал.

Студенти можуть редагувати особисті дані, переглядати та записуватися на курси, проходити тести та опитування, спілкуватися через чати, форуми, повідомлення, мати доступ до груп та файлообмінника, виконувати пошук.

Викладачі можуть створювати курси, визначати права доступу, створювати тести та навчальні матеріали, керувати ними, переглядати статистику використання. Також вони можуть керувати електронною поштою курсу, створювати оголошення, опитування.

Адміністратори мають повноваження керувати користувачами системи та їхніми правами, керувати курсами, резервними копіями, налаштовувати загальні параметри, такі як теми оформлення, мова інтерфейсу тощо.

Однак, оскільки ці системи встановлюються як окремі веб-сайти, це не завжди зручно для навчальних закладів. Тому варто розглянути інтеграцію систем керування навчанням із сучасними системами керування контентом.

Для інтеграції системи керування навчанням (LMS) з популярними системами керування контентом (CMS) існують різні рішення.

Для WordPress найкращою LMS є Namaste! LMS. Для Drupal таку функцію виконує Adrenna Academic. Але найбільше розширень для інтеграції LMS доступно для CMS Joomla!

Серед них – Guru, Joomlalearn LMS Basic Edition (рис. 1.3), Joomla! та Joomla-Reports. Перші три є повноцінними LMS, які крім збереження оцінок виконують ще низку функцій (зберігання навчальних матеріалів, контроль переглядів тощо). Однак вони створюють додаткове навантаження на сервер, займають багато місця і генерують багато таблиць у базі даних, що може бути проблемою. Guru також є платним рішенням за \$297.

					<i>КР. КН 24.552.04.000 ПЗ</i>	Арк.
						13
Змн.	Арк.	№ докум.	Підпис	Дата		

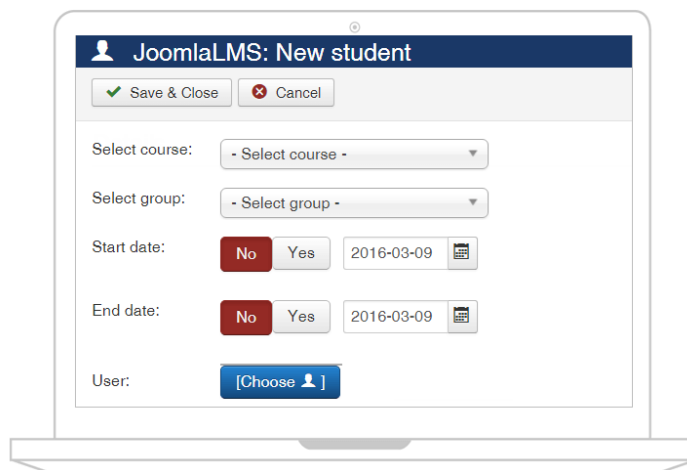


Рисунок 1.3 – JoomlaLMS Basic Edition

Найближчим до бажаного функціоналу є Joom-Reports, але воно не дозволяє розмежувати права викладачів за предметами.

Отже, потрібне програмне забезпечення, яке дозволить працювати з оцінками учнів/студентів за допомогою веб-інтерфейсу без створення великого навантаження на сервер та інтегруватиметься з однією з популярних CMS. Таке рішення було б дуже корисним, зокрема для Галицького фахового коледжу.

1.3 Аналіз вимог до програмного засобу та постановка завдання

На основі аналізу потреб навчальних закладів та залучених сторін можна визначити кілька ключових вимог до розроблюваного веб-додатку.

Застосунок повинен поділятися на адміністративну та публічну (фронтальну) частини з різними рівнями доступу та функціоналом.

Публічна частина має містити інтерфейси для студентів та викладачів з відповідними можливостями перегляду, додавання та редагування оцінок.

Доступ до публічної частини можливий лише після внесення початкових даних адміністратором (навчальні групи, студенти, викладачі, предмети). Перш ніж студенти та викладачі зможуть розпочати роботу з публічною (фронтальною) частиною системи, адміністратор повинен буде внести базову інформацію через адміністративну частину. Ця інформація

					КР. КН 24.552.04.000 ПЗ	Арк.
						14
Змн.	Арк.	№ докум.	Підпис	Дата		

включатиме дані про навчальні групи, списки студентів та викладачів, а також перелік предметів, що вивчаються.

Адміністративна частина повинна забезпечувати:

- внесення початкових даних про навчальні групи, студентів, викладачів, предмети;
- збереження даних у базі даних;
- редагування та видалення існуючих даних;
- додавання, редагування та видалення оцінок студентів.

Адміністратор, який матиме повний доступ до налаштувань та управління системою через адміністративну частину, не зможе взаємодіяти з функціоналом публічної частини.

Інтерфейс викладача у публічній частині має надавати можливості:

- вносити оцінки для відповідних навчальних груп та предметів;
- редагувати раніше внесені некоректні оцінки;
- зберігати внесені зміни.

Інтерфейс студента у публічній частині системи буде призначений виключно для перегляду поточних оцінок. Студенти не матимуть можливості редагувати чи видаляти свої оцінки через цей інтерфейс.

Інтерфейси як адміністративної, так і публічної частин системи будуть розроблені з урахуванням принципів зручності використання та інтуїтивного розуміння. Це забезпечить легкість опанування та ефективну роботу для всіх категорій користувачів: адміністраторів, викладачів та студентів.

Для забезпечення сумісності та можливості інтеграції з іншими системами, а також для полегшення майбутніх розширень функціоналу, система має бути інтегрована з популярною системою керування контентом Joomla!.

					<i>КР. КН 24.552.04.000 ПЗ</i>	Арк.
						15
Змн.	Арк.	№ докум.	Підпис	Дата		

2 ПРОЄКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

2.1 Проєктування функціонування інформаційної системи

Застосунок поділяється на дві частини: адміністративну і фронтальну. Фронтальна частина також поділяється на два елементи: інтерфейс для студента та для викладача.

Робота із фронтальною частиною сайту не може бути розпочата до того часу, поки адміністратор не заповнить систему інформацією.

Роботу адміністративної частини додатку можна поділити на такі етапи:

- Внесення інформації про навчальні групи, студентів, викладачів, предмети. Це перший етап роботи після встановлення системи. На цьому етапі відбувається наповнення її інформацією необхідною для внесення оцінок із фронтальної частини.

- Запис інформації у базу даних. Усі дані, отримані в попередньому кроці зберігаються в базі даних сайту, що дозволяє комфортно працювати з ними за допомогою мови запитів SQL.

- Внесення нової інформації, редагування, або видалення наявної інформації. За потреби інформація про навчальні групи, студентів, викладачів, предмети може бути відредагована адміністратором. Також адміністратор завжди може додати нові об'єкти в систему.

- Внесення нових оцінок, редагування, видалення уже існуючих. Адміністративна частина додатку дає змогу вносити нові оцінки, але для цього призначена перш за все фронтальна частина системи. Окрім цього адміністратор має змогу редагувати та видаляти уже існуючі оцінки. Цей функціонал призначений для виправлення помилок, допущених при внесенні інформації із фронтальної частини додатку.

Алгоритм роботи адміністративної частини системи представлено на рисунку 2.1.

					КР. КН 24.552.04.000 ПЗ	Арк.
						16
Змн.	Арк.	№ докум.	Підпис	Дата		

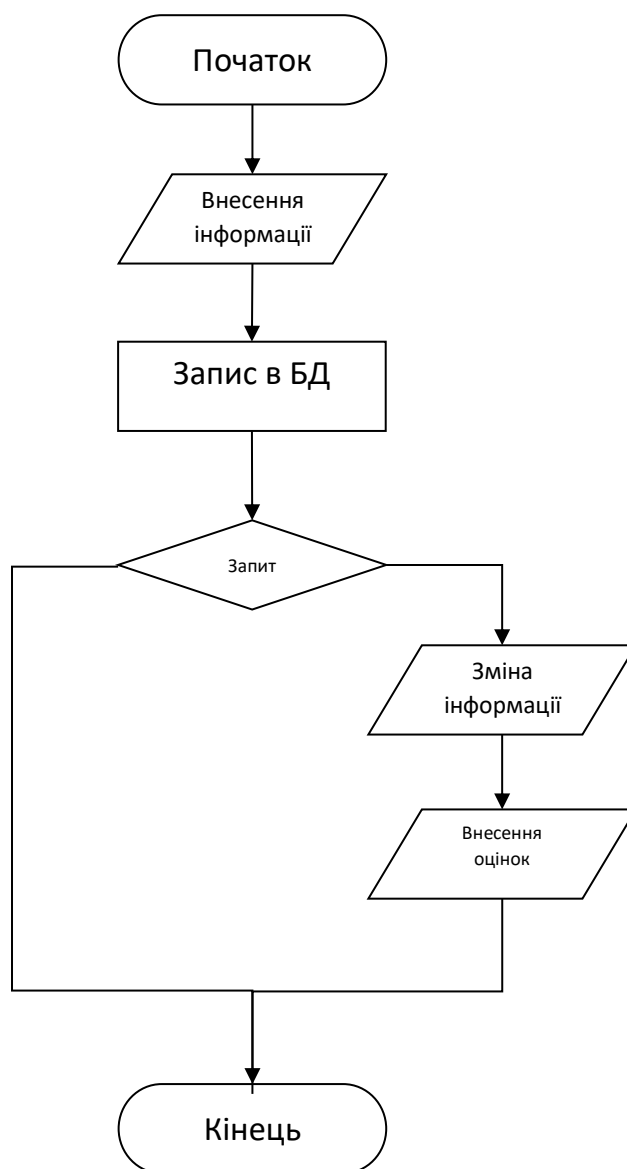


Рисунок 2.1 – Алгоритм роботи адміністративної частини системи

Робота фронтальної частини компоненту викладача поділяється на такі кроки:

- внесення оцінок;
- редагування незадовільних оцінок;
- збереження результатів.

Робота фронтальної частини компоненту для студента включає в себе лише виведення оцінок. Діаграми варіантів використання представлені в додатку Б.

2.2 Проєктування інтерфейсу

Інтерфейс майже будь-якого компонента Joomla! поділяється на адміністративну і фронтальну частини.

Окрім того інтерфейс фронтальної частини сайту компонента буде відрізнятися для учнів і викладачів.

Схема поділу інтерфейсу компонента подана на рисунку 2.2.

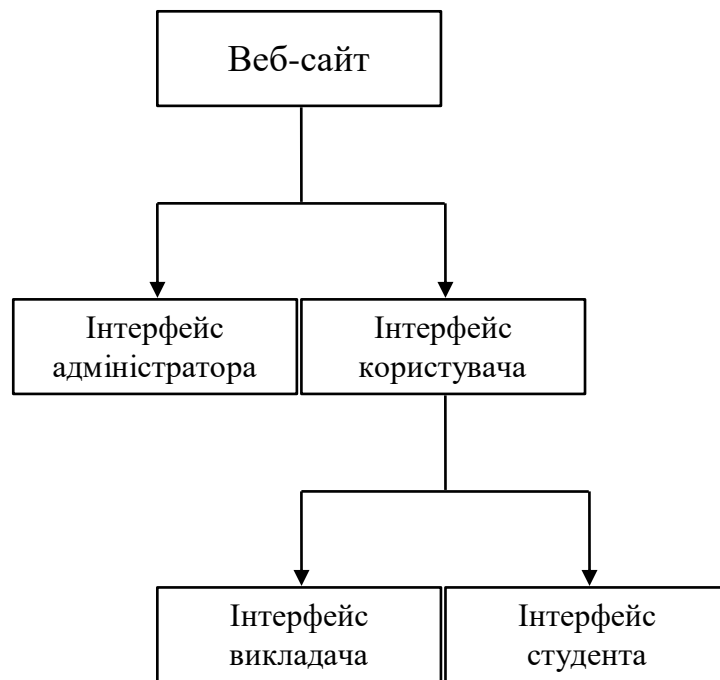


Рисунок 2.2 – Схема поділу інтерфейсу компонента

У фронтальній частині проекту необхідно надати доступ до інформації двом групам користувачів – студентам та викладачам. Визначення належності до певної групи відбувається на основі ідентифікатора користувача, який записується в таблиці бази даних сайту на CMS Joomla! і в відповідній (students, або teachers) таблиці бази даних компонента.

Залежно від належності до певної групи користувачів буде відрізнятися і інтерфейс.

Для студентів необхідно надати лише інформацію про їх оцінки, без можливості редагування. Макет інтерфейсу для студента подано на рисунку 2.3.

Мітка		
Назва предмету	Викладач	Оцінка
Інформація		

Рисунок 2.3 – Інтерфейс фронтальної частини компонента для студента

Виведення цієї інформації відбувається при переході за посиланням, або при виборі пункту меню.

Для викладачів необхідно передбачити можливість перегляду, редагування і додавання нових оцінок. Але лише для тих груп і студентів у яких вони викладають.

Після входу у компонент викладач отримує список предметів, які він викладає. Наступним кроком після вибору предмету є вибір навчальної групи, оцінки студентів якої хоче внести викладач. Після вибору групи викладач отримує можливість ввести чи редагувати оцінки. Макет форми, яка дозволяє проводити ці операції, відображена на рисунку 2.4.

Мітка		
Ім'я студента	Прізвище студента	Оцінка
Кнопка збереження		

Рисунок 2.4 – Інтерфейс фронтальної частини компонента для викладача

Таблиця заповнюється інформацією із бази даних. Якщо оцінки для певного студента ще немає, то виводиться клітинка із символом “-“. Оцінки, відмінні від 2, не доступні для редагування.

При натисненні кнопки “Зберегти” відбувається оновлення, або запис нової інформації (залежно від наявності оцінки) в базу даних.

2.3 Проєктування адміністративної частини

Адміністратор може виконувати такі дії у компоненті:

- перегляд, додавання і видалення студентів;
- перегляд, додавання, редагування і видалення навчальних груп;
- перегляд, додавання, редагування і видалення викладачів;
- перегляд, додавання, редагування і видалення предметів;
- перегляд, додавання, редагування і видалення оцінок.

Для кожної із цих функцій, крім видалення, яке відбувається із форми перегляду, необхідно передбачити окрему форму.

Форми редагування і додавання об’єднано в одну. Різниця між ними буде лише у тому, що при редагуванні в форму передаються певні дані за наперед визначеним ідентифікатором, а при додаванні нового запису він буде відсутній.

Усі форми, які представляють списки об’єктів оснащені кнопками створення нового запису, редагування виділеного запису і кнопкою видалення запису. Якщо виділити кілька записів і спробувати їх відредагувати, відбудеться редагування першого із них. При спробі видалити кілька записів, відбувається видалення усіх виділених записів.

Усі форми призначені для редагування записів чи створення нових оснащені кнопками “Зберегти”, “Зберегти і закрити” і “Скасувати”.

При натисненні кнопки “Зберегти і закрити” відбувається оновлення, або внесення даних у базу даних (залежно від задачі, яка виконується) і

					<i>КР. КН 24.552.04.000 ПЗ</i>	Арк.
						20
Змн.	Арк.	№ докум.	Підпис	Дата		

перенаправлення адміністратора на сторінку перегляду списку відповідних об'єктів.

При натисненні кнопки “Зберегти” відбувається оновлення, або внесення даних у базу даних (залежно від задачі, яка виконується). Користувач залишається на тій же сторінці.

При натисненні кнопки “Скасувати” відбувається перенаправлення адміністратора на сторінку перегляду списку відповідних об'єктів без будь-яких змін у базі даних.

Усі форми мають лише “статичні” елементи. Тобто завантаження даних у їх елементи відбувається один раз – при завантаженні сторінки. Тільки форма додавання оцінок має бути оснащена динамічними зв'язаними списками. Тобто при виборі викладача в список предметів завантажуються лише ті предмети, які він викладає. Після вибору предмету у список груп завантажуються лише ті групи, у яких читається цей предмет. Після вибору групи у список студентів завантажуються лише студенти цієї групи.

Макет форм додавання і редагування об'єктів адміністративної частини відображено на рисунку 2.5.

Панель інструментів	Кнопки
Робоча зона	

Рисунок 2.5 – Макет форм додавання інформації

Макет форм перегляду списків подано на рисунку 2.6.

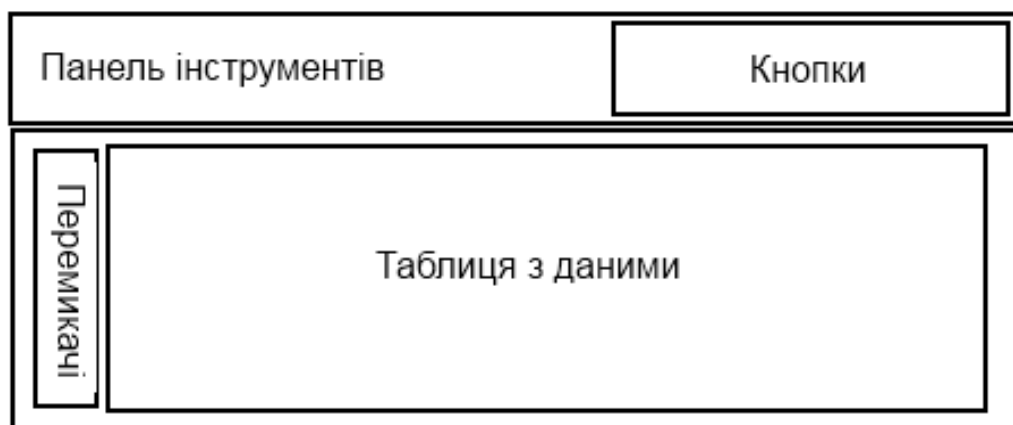


Рисунок 2.6 – Макет форм перегляду інформації

Адміністративна частина надає функціонал для перегляду, додавання, редагування та видалення студентів, груп, викладачів, предметів та оцінок. Форми редагування та додавання об'єднано. Є кнопки створення, редагування та видалення записів. При спробі редагувати/видалити кілька записів, діятиме на перший/всі виділені відповідно. Форми мають кнопки "Зберегти", "Зберегти і закрити" та "Скасувати". Форма додавання оцінок має динамічні зв'язані списки для вибору об'єктів. Наведено макети форм.

2.4 Проектування бази даних

База даних веб-сайту призначена для зберігання усього обсягу інформації, яка у ньому використовується. Компонент, який розробляється, не є винятком, тому усі його дані також будуть зберігатися у БД. В результаті аналізу предметної області було виділено наступні сутності:

- студент;
- викладач;
- група;
- предмет;
- оцінка.

Для їх збереження у БД необхідно створити такі таблиці:

- students;

- teachers;
- groups;
- subjects;
- marks;
- subj_to_group.

Таблиця `subj_to_group` є додатковою таблицею, яка дозволить реалізувати зв'язок “багато до багатьох” між таблицями предметів і груп.

Зв'язки між таблицями бази даних компонента і базою даних системи керування контентом показано на рисунку 2.7.

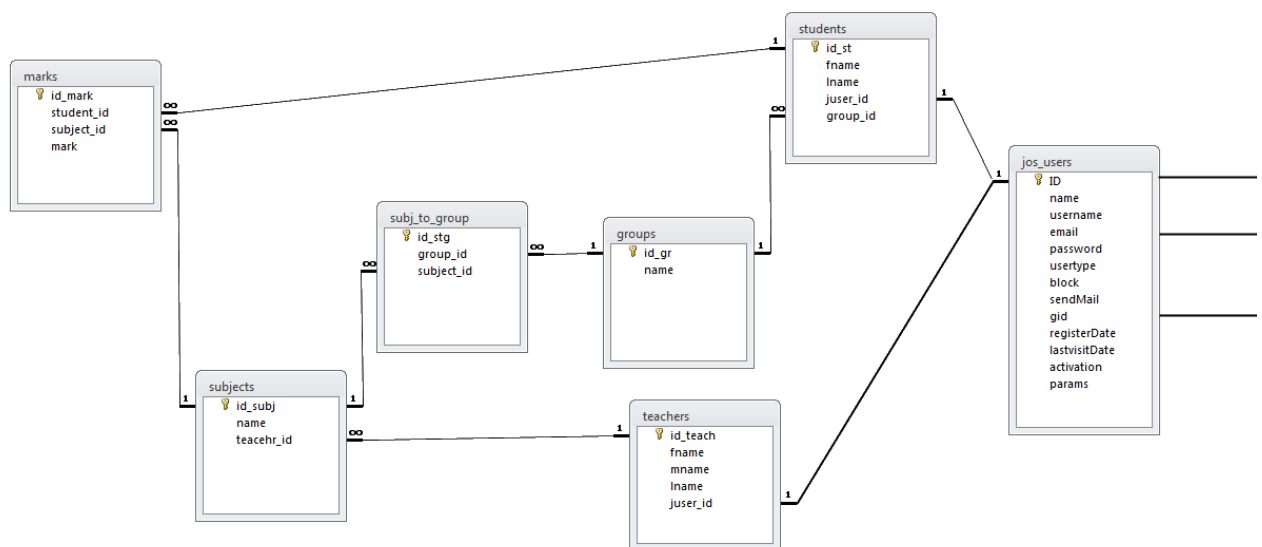


Рисунок 2.7 – Зв’язок таблиць БД компоненту і CMS

Таблиця `groups` призначена для зберігання назви і ідентифікатора групи. Структура таблиці зображена на рисунку 2.8.

	Field Name	Data Type
🔑	id_gr	AutoNumber
	name	Text

Рисунок 2.8 – Структура таблиці `groups`

Таблиця `marks` призначена для зберігання значення оцінки, її ідентифікатора, ідентифікатора студента, якому виставлена ця оцінка і

ідентифікатора предмету, з якого виставлена ця оцінка. Словник таблиці відображений у рисунку 2.9.

	Field Name	Data Type
🔑	id_mark	AutoNumber
	student_id	Number
	subject_id	Number
	mark	Number

Рисунок 2.9 – Структура таблиці marks

Таблиця students призначена для зберігання імені та прізвища студента, його ідентифікатора, ідентифікатора користувача Joomla!, який пов’язується із студентом і ідентифікатора групи, в якій навчається студент. Типи і назви полів таблиці зображені на рисунку 2.10.

	Field Name	Data Type
🔑	id_st	AutoNumber
	fname	Text
	lname	Text
	juser_id	Number
	group_id	Number

Рисунок 2.10 – Структура таблиці students

Таблиця subject_to_group не представляє жодну сутність предметної області, а служить лише для реалізації зв'язку “багато до багатьох” між таблицями subjects і groups. Структура таблиці відображена на рисунку 2.11.

	Field Name	Data Type
🔑	id_stg	AutoNumber
	group_id	Number
	subject_id	Number

Рисунок 2.11 – Структура таблиці subj_to_group

Таблиця subjects призначена для зберігання назви предмета, його ідентифікатора і ідентифікатора викладача, який викладає цю дисципліну. Словник таблиці відображений на рисунку 2.12.

	Field Name	Data Type
🔑	id_subj	AutoNumber
	name	Text
	teacher_id	Number

Рисунок 2.12 – Структура таблиці subjects

Таблиця teachers призначена для зберігання прізвища, імені, по-батькові викладача, його ідентифікатора і ідентифікатора користувача Joomla!, з яким буде пов'язуватися викладач. Структура таблиці зображена на рисунку 2.13.

	Field Name	Data Type
🔑	id_teach	AutoNumber
	fname	Text
	mname	Text
	lname	Text
	juser_id	Number

Рисунок 2.13 – Структура таблиці teachers

Для зберігання даних компонента в базі даних виділено такі сутності: студент, викладач, група, предмет, оцінка. Для них створено відповідні таблиці students, teachers, groups, subjects, marks. Додатково створено таблицю subj_to_group для реалізації зв'язку "багато до багатьох" між предметами та групами.

Таблиці groups і subjects зберігають назви груп і предметів відповідно. Таблиця marks зберігає оцінки та ідентифікатори студента, предмета. Таблиця students зв'язує студентів з групами та користувачами Joomla. Таблиця teachers зв'язує викладачів з предметами та користувачами Joomla.

3 РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ

3.1 Обґрунтування технологій і засобів реалізації

Оскільки доцільною є інтеграція системи із існуючими веб-сайтами, то потрібно розробляти її з врахуванням особливостей певної сучасної системи керування контентом. Для цієї ролі обрано CMS Joomla!.

Joomla – це система управління вмістом (CMS), що дає змогу створювати вебсайти і потужні онлайн-застосунки. Вона неодноразово отримувала нагороди як найкраща безкоштовна CMS і на даний момент має одне з найбільш активних спільнот серед безкоштовних CMS. Joomla набула популярності завдяки простоті використання, розширюваності і ще безлічі чинників. На сьогодні Joomla є однією з найпопулярніших CMS в мережі інтернет, адже на базі Joomla працює 2,8% всіх web-сайтів.

Головні її переваги це:

- Легкість у використанні. Joomla має вбудований редактор для додавання інформації, схожий на Microsoft Word. Він зручний і легкий в освоєнні. Крім того в розпорядженні власника сайту є зручна адміністраторська панель, яка дозволяє легко керувати веб-сторінками.
- Розширюваність. Функціонал сайтів, створених на базі Joomla CMS, може бути легко доповнений. Існує величезна кількість (більше 9000) розширень для Joomla, значна частина з яких є безплатними. Таким чином, удосконалення сайту не займе багато часу і коштів.
- Гнучкий оформлення. Joomla дозволяє використовувати декілька різних дизайнів сайту. Тому, за бажанням, зміна дизайну не торкнеться вмісту сайту і відбудеться майже непомітно. Крім того механізм швидкої зміни дизайну дозволяє розробляти різні варіанти оформлення для свят або пір року.
- Стабільність. У Joomla найбільша кількість користувачів і розробників (більше 200 тисяч по всьому світу) і кілька перемог у номінації «краща CMS року», остання з яких отримана в 2021 році. Регулярне

					КР. КН 24.552.04.000 ПЗ	Арк.
						26
Змн.	Арк.	№ докум.	Підпис	Дата		

оновлення системи забезпечує актуальність проєкту і задоволення потреб спільноти.

Окрім цього, Joomla має відкритий код, докладну документацію і потужний фреймворк, що є дуже зручним для створення розширень на її базі.

Будь-який веб-застосунок базується на таких складових:

- серверна мова програмування;
- СУБД;
- HTML-розмітка;
- каскадні таблиці стилів (CSS);
- JavaScript.

В якості серверної мови програмування доцільно обрати PHP, адже саме з її допомогою створено Joomla CMS, а отже ніяких конфліктів при інтеграції створюваного розширення і системи керування вмістом не буде.

PHP – скриптова мова програмування, створена для генерації HTML-сторінок на стороні веб-сервера, розробляється і підтримується групою ентузіастів і поширюється за власною, не сумісною із GNU/GPL, ліцензією. Вона володіє C-подібним синтаксисом, що спрощує її вивчення. PHP є дуже гнучкою мовою і має інтерфейси для роботи із багатьма СУБД (MySQL, PostgreSQL, mSQL, Oracle, dbm, Hyperware, Informix, InterBase, Sybase). Окрім генерації розмітки, існує можливість генерувати JavaScript-код, що збільшує інтерактивність сторінок. Завдяки своїй інтерпритованості, PHP є дуже ефективною при використанні в малих і середніх веб-додатках. Але при використанні в великих високонавантажених системах, де швидкість роботи є критичною, доцільніше використовувати компільовані мови програмування.

У CMS Joomla! використовується шаблон програмування Model-View-Controller (MVC). Цей шаблон має на меті забезпечити гнучкий дизайн програмного забезпечення, що полегшує подальші зміни чи розширення програм, а також надає можливість повторного використання окремих компонентів. У великих системах використання MVC призводить до певної

					<i>КР. КН 24.552.04.000 ПЗ</i>	Арк.
						27
Змн.	Арк.	№ докум.	Підпис	Дата		

впорядкованості структури і робить її зрозумілішою завдяки зменшенню складності.

Основна ідея MVC – відокремлення даних (модель) від інтерфейсу користувача (вигляду) таким чином, щоб зміни інтерфейсу мінімально впливали на роботу з даними, а зміни в моделі даних могли здійснюватися без змін інтерфейс[1].

Шаблон MVC поділяє систему на три ключові частини:

- Модель даних - відповідає за зберігання даних та правила їх обробки.
- Вигляд даних - відповідає за представлення даних користувачеві.
- Контролер - керує взаємодією між моделлю та виглядом, а також взаємодією із зовнішніми елементами, такими як користувацькі запити.

Взаємодія цих трьох частин між собою і з зовнішніми елементами представлена на рисунку 3.1.

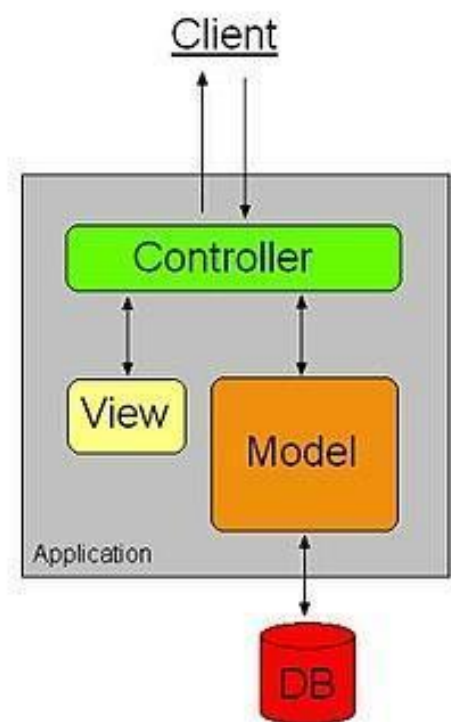


Рисунок 3.1 – Взаємодія складових частин шаблону MVC

HTML – це мова розмітки, яка використовується для створення веб-сторінок. Коли браузер відкриває таку сторінку, він читає її HTML-код та інтерпретує спеціальні символи (теги) як інструкції для відображення різних елементів, таких як зображення, таблиці, посилання тощо.

Для керування зовнішнім виглядом веб-сторінок, створених за допомогою HTML, використовуються каскадні таблиці стилів (CSS). CSS - це формальна мова, яка описує візуальне представлення документів, написаних мовами розмітки. Вона найчастіше застосовується для оформлення веб-сторінок HTML та XHTML, але також може використовуватися для будь-яких XML-документів, наприклад, SVG або XUL.

Для роботи з базами даних було обрано систему управління базами даних MySQL [2]. MySQL – це вільна реляційна СУБД, розробку та підтримку якої здійснює корпорація Oracle. Ця система є безкоштовною, швидкою та стійкою. Вона має відкритий вихідний код і надає програмний інтерфейс (API) для взаємодії з великою кількістю мов програмування, включаючи PHP.

Для підвищення інтерактивності веб-сторінок використовується мова програмування JavaScript. JavaScript є об'єктно-орієнтованою мовою, але використання в ній прототипування призводить до відмінностей у роботі з об'єктами порівняно з традиційними класовими мовами програмування. Крім того, JavaScript має ряд властивостей, характерних для функціональних мов, таких як функції як об'єкти першого класу, об'єкти як списки, каррінг, анонімні функції, замикання, що додає мові додаткової гнучкості.

JavaScript має синтаксис, схожий на C, але має суттєві відмінності:

- об'єкти з можливістю інтроспекції;
- функції розглядаються як об'єкти першого класу;
- автоматичне приведення типів даних;
- автоматичне "прибирання сміття" (garbage collection);
- підтримка анонімних функцій.

В JavaScript відсутні такі корисні речі, як:

- модульна система;
- стандартна бібліотека;
- стандартні інтерфейси до веб-серверів і баз даних;

За допомогою JavaScript можна реалізувати технологію AJAX (Asynchronous Javascript and XML). AJAX дозволяє здійснювати "фоновий" обмін даними між браузером і веб-сервером. Це дає можливість оновлювати дані на веб-сторінці без її повного перезавантаження, що робить веб-додатки швидшими і зручнішими [3].

Використання AJAX значно скорочує трафік, оскільки часто потрібно завантажити лише ту частину сторінки, яка змінилася, а не всю сторінку повністю. Це також знижує навантаження на сервер і прискорює реакцію користувацького інтерфейсу, адже користувач бачить результати своїх дій швидше.

У системі, що розробляється, AJAX може бути використаний для динамічного формування списків, запису даних у базу даних тощо.

3.2 Реалізація інтерфейсу користувача

У CMS Joomla! генерація елементів інтерфейсу відбувається за допомогою двох основних класів, які є у кожному додатку: HTML_gradebook і TOOLBAR_gradebook.

Методи класу HTML_gradebook отримують в якості параметрів інформацію і виводять її на екран.

Методи класу TOOLBAR_gradebook формують кнопки на панелі інструментів [4].

Для реалізації адміністративної частини додатку потрібно створити такі форми:

- перегляд студентів;
- додавання і видалення студентів;
- редагування і додавання навчальних груп;

					КР. КН 24.552.04.000 ПЗ	Арк.
						30
Змн.	Арк.	№ докум.	Підпис	Дата		

- перегляд навчальних груп;
- перегляд викладачів;
- додавання, редагування викладачів;
- перегляд предметів;
- додавання і редагування предметів;
- перегляд оцінок;
- їх додавання і редагування.

Кожний із цих форм відповідає окремий метод класу HTML_gradebook і метод класу TOOLBAR_gradebook.

Усі форми перегляду мають типову структуру. Їх реалізацію розглянуто на прикладі форми перегляду навчальних предметів. Метод, який відповідає за цю форму генерує html-код форми, яка містить у собі таблицю, в першій колонці якої розміщуються прапорці для групового виділення, а в інших – інформацію про предмети.

Також форма має три приховані поля, які зберігають значення змінних option і task, необхідних для коректної роботи системи, і ідентифікатори виділених рядків таблиці.

Генерація прапорців відбувається методом grid.id класу JHTML із фреймворку Joomla. В лістингу 3.1 наведено програмну реалізацію виводу прапорців у формі.

Лістинг 3.1 – вивід прапорців у формі

```
$k = 0;
    for ($i = 0, $n = count($rows); $i < $n; $i++)
    {
        $row = &$rows[$i];
        $checked = JHTML::_('grid.id', $i, $row-
>id_subj);
        ...
        <td>
            <?=$checked?>
        </td>
        ...
    }
```


Уся інформація у таблиці подається у вигляді тексту з посиланням. Посилання веде на форму редагування предмету [5]. Реалізація виведення одного рядка таблиці подана в лістингу 3.2.

Лістинг 3.2 – Виведення рядка таблиці

```
<td align=center><?='<a href="'. $link.'">' . $row->name.'</a>'
?> </td>
<?php
$query = 'SELECT lname, fname, mname FROM
#__gradebook_teachers WHERE id_teach = ' . $row->teacher_id;
$db->setQuery($query);
$teacher_name = $db->loadRow();
$teacher_name= $teacher_name[0].' '.substr($teacher_name[1],
0,2).'.'.substr($teacher_name[2],0,2).'.';
?>
<td align=center><?='<a href="'. $link.'">' . $teacher_name.'
</a>' ?> </td>
<?php
$query = 'SELECT group_id FROM #__gradebook_subj_to_group
WHERE subject_id = ' . $row->id_subj;
$db =& JFactory::getDBO();
$db->setQuery($query);
$groups_ids_array = $db->loadResultArray();
$groups_ids_string="";
for($j=0;$j<count($groups_ids_array);$j++){
if($j!=0)$groups_ids_string=$groups_ids_string.',' . $groups_i
ds_array[$j];
else
$groups_ids_string = $groups_ids_string.$groups_ids_
array[$j];
}
?>
<td align=center><?='<a href="'. $link.'">' . $groups_ids_
string .' </a>' ?> </td>
```

Вигляд реалізованої форми подано на рисунку 3.2.

	Назва предмету	Викладач предмету	Навчальні групи
<input type="checkbox"/>	Математика	Кут Іван Іванович	КН-11
<input type="checkbox"/>	Біологія	Кут Іван Іванович	КН-21
<input type="checkbox"/>	Фізика	Дудар Ольга Олегівна	КІ-21
<input type="checkbox"/>	Історія України	Дудар Ольга Олегівна	КН-21

Рисунок 3.2 – Форма перегляду навчальних предметів

Кнопки “Створити”, “Редагувати” і “Знищити” реалізовані методом _DEFAULT_SUBJECT класу TOOLBAR_gradebook. В лістингу 3.3 наведено їх реалізацію [6].

Лістинг 3.3 – Код кнопок

```
function _DEFAULT_SUBJECTS() {
    JToolBarHelper::title
(JText::_('COM_GRADEBOOK_TOOLBAR_SUBJECTS'), 'generic.png');
    JToolBarHelper::addNew('new_subject');
    JToolBarHelper::editList('new_subject', 'COM_GRADEBOOK_EDIT');
;
    JToolBarHelper::deleteList(JText::_('COM_GRADEBOOK_TOOLBAR_R
EMOVE_SUBJECTS_CONFIRMATION'), 'remove_subjects');
}
```

Інші форми перегляду інформації реалізуються за тим самим принципом, але відрізняються інформацією, яка виводиться.

Крім цих форм необхідно також створити форми для додавання інформації. Реалізацію цього типу форм розглянуто на прикладі форми створення нового предмету. За це відповідає метод createNewSubjet.

Для генерації елементів форми використовується статичний метод класу JHTML. В лістингу 3.4 наведено приклад реалізації вибірки даних про викладачів і реалізації їх списку з його допомогою.

Лістинг 3.4 – Код реалізації вибірки даних про викладачів і реалізації їх.

```
$query = 'SELECT id_teach,lname FROM #__gradebook_teachers';
$db =& JFactory::getDBO();
$db->setQuery($query);
$teachers_array = $db->loadObjectList();

echo JHTML::_('select.genericlist', $teachers_array,
'teacher_id', 'size=25', 'id_teach', 'lname', $row->teacher_id);
```

Метод JHTML::_ отримує в якості параметрів тип елемента, який необхідно згенерувати, масив об’єктів, на основі якого генерується елемент форми, атрибут об’єкту який використовується як ідентифікатор елементу форми, атрибут об’єкту який відображається на формі і (тільки для списків) номер виділеного по замовчуванню об’єкта [7].

					КР. КН 24.552.04.000 ПЗ	Арк.
						33
Змн.	Арк.	№ докум.	Підпис	Дата		

Результат роботи коду, приведеного вище, зображений на рисунку 3.3.

Рисунок 3.3 – Згенерований елемент форми

Тим же методом реалізуються і інші елементи форми. Панель управління, як і в формах перегляду інформації, створюється методом класу HTML_gradebook [8]. В лістингу 3.5 наведено код реалізації методу для цієї форми.

Лістинг 3.5 – Код реалізації методу для форми

```
function _NEW_SUBJECT()
{
    JToolBarHelper::title(JText::_('COM_GRADEBOOK_TOOLBAR_ADD_SUBJECT'), 'generic.png');
    JToolBarHelper::apply('apply_subject');
    JToolBarHelper::save('save_subject');
    JToolBarHelper::cancel('subjects');
}
```

Результатом поєднання цих методів буде готова форма, вигляд якої зображено на рисунку 3.4.

Додати дисципліну

[Зберегти](#)[Зберегти і закрити](#)[Скасувати](#)

Назва предмету:

Викладач:

Кут Іван Іванович
Дудар Ольга Олегівна

Група:

КН-11
КН-21
КІ-11
КІ-21

Рисунок 3.4 – Форма додавання дисциплін

Інші форми додавання інформації реалізуються за тим же принципом, але відрізняються кількістю і призначенням елементів.

Клас `HTML_gradebook` оголошений в файлі `admin.gradebook.html.php`. Клас `TOOLBAR_gradebook` оголошений в файлі `toolbar.gradebook.php.html`.

Фронтальна частина додатку виводить усі оцінки студента, або список предметів і груп, в яких працює викладач, залежно від ролі користувача в системі.

Інтерфейс для студентів реалізується методом `showMarks` методу `HTML_gradebook`. Викладацька частина реалізується методами `showSubjects`, `showGroups` і `showStudents`. При вході у застосунок викликається метод `showSubjects`, який дозволяє викладачу обрати предмет, з якого він бажає виставити оцінки[9].

Після вибору предмету, метод `showGroups` демонструє список груп, в яких викладається обраний раніше предмет.

Наступним кроком є виведення списку студентів і їх оцінок у форму методом `showStudents`. Вигляд цієї форми зображено на рисунку 3.5.

					<i>КР. КН 24.552.04.000 ПЗ</i>	Арк.
						35
Змн.	Арк.	№ докум.	Підпис	Дата		

Додати оцінку
Зберегти
Зберегти і закрити
Скасувати

Предмет:

КН-11 - Математика (Кут Іван Іванович)
КН-21 - Біологія (Кут Іван Іванович)
КІ-21 - Фізика (Дудар Ольга Олегівна)
КН-21 - Історія України (Дудар Ольга Олегівна)

Студент:

Козак Іван (КН-11)

Оцінка:

Рисунок 3.5 – Форма внесення оцінок

Клас `HTML_gradebook` для фронтальної частини описаний в файлі `gradebook.html.php`.

3.3 Реалізація функціоналу

До основних функцій системи належать запис даних про викладачів, предмети, групи і студентів в базу даних, виведення інформації про оцінки студента, а також генерація звітів про успішність груп.

Для реалізації запису інформації в базу даних було створено 6 класів для роботи з таблицями бази даних (моделей в термінології MVC):

- `TableGroup`;
- `TableStudent`;
- `TableTeacher`;
- `TableMark`;
- `TableSubject`;
- `TableLink`.

Кожен із цих класів розширює клас фреймворку Joomla `JTable`. Запис даних в таблиці відбувається за допомогою методу `bind` цих класів. Варто зауважити, що, для коректної роботи цього методу, назви елементів форм, з яких зчитується інформація, і назви полів у базі даних мають збігатися [10].

Якщо ж дані потрібно записати у кілька таблиць варто використати клас JFactory для підключення до бази даних і виконання запитів. Щоб навести приклад таких дій, представлено лістинг 3.6 запису в базу даних інформації про оцінки студентів:

Лістинг 3.6 –Запис даних у кілька таблиць

```
$mainframe =& JFactory::getApplication('site');
$students_marks=$_POST["_studentsMarks"];
for($i=0;$i<count($students_marks);$i=$i+6){
    if($students_marks[$i+1]!=0){
        $db =& JFactory::getDbo();
        $query = "UPDATE #__gradebook_marks SET mark
=".$students_marks[$i+4]."" WHERE id_mark
=".$students_marks[$i+1];
        $db->setQuery($query);
        $result = $db->query();
    }
    else{
        if($students_marks[$i+4]!='-'){
            $db =& JFactory::getDbo();
            $query = "INSERT INTO #__gradebook_marks
(student_id, subject_id, mark) VALUES (". $students_marks[$i].",
".$students_marks[$i+5].", ".$students_marks[$i+4].")";
            $db->setQuery($query);
            $result = $db->query();
        }
    }
}
```

Виведення інформації про оцінки студента відбувається методом showMarks. При його виклику відбувається перевірка ролі користувача в системі. Для визначення користувача використовується метод getUser класу JFactory. Після цього, якщо користувач увійшов на сайт із використанням свого логіну, визначається ідентифікатор студента. За цим ідентифікатором відбувається вибірка даних із БД. Якщо користувач виявиться викладачем, то він зможе побачити лише незаповнену сторінку. У випадку ж, якщо система визначить користувача як студента, то відбудеться виведення його оцінок, за допомогою методу HTML_gradebook.

Ще однією функцією системи є генерація звітів. Реалізується вона в окремому PHP-файлі з назвою report.php. При його виклику відбувається

					КР. КН 24.552.04.000 ПЗ	Арк.
						37
Змн.	Арк.	№ докум.	Підпис	Дата		

оголошення системних констант Joomla!, отримання посилання на клас Jfactory та визначення мови користувача. Наступним кроком є отримання змінної із значенням ідентифікатора групи, для якої формується звіт, із масиву GET.

З допомогою отриманого значення з бази даних вибираються дані про студентів групи і предмети, які в ній викладаються. Після цього в циклі будується таблиця. При кожній ітерації циклу виводиться один рядок таблиці, що відповідає даним про одного студента.

3.4 Тестування

Тестування інформаційної системи проведено згідно такого плану:

- перевірка запису інформації в БД з адміністративної частини;
- перевірка виведення інформації з БД в адміністративній частині;
- перевірка формування звітів;
- перевірка введення оцінок із фронтальної частини;
- перевірка виведення інформації у фронтальній частині.

Для перевірки запису інформації в БД відкрито форму додавання нової групи (рис. 3.6), заповнено її інформацією і натиснуто кнопку “Зберегти і закрити”.

Рисунок 3.6 – Форма додавання нової групи

Щоб переконатися в тому, що запис пройшов успішно, відкрито веб-застосунок phpMyAdmin, який є графічною оболонкою для СУБД MySQL, і переглянуто таблицю #_gradebook_groups із бази даних сайту (рис. 3.7).







+ Options				
			id_gr	name
			1	K-47
			30	K-17

Рисунок 3.7 – Результат запису інформації

Протестовано форму додавання студента (рис. 3.8). Внесено ім'я та прізвище студента, вибрано користувача Joomla, який буде асоціюватися із цим студентом та групу, в якій навчається студент.

Додати студента Зберегти Зберегти і закрити Скасувати

Студент: Прізвище студента:

Логін: Пароль:

Група:

- КН-11
- КН-21
- КІ-11
- КІ-21

Рисунок 3.8 – Форма додавання студентів

Результат запису інформації про студента зберігається в таблиці #_gradebook_students (рис. 3.9).







+ Параметри									
				id_teach	ffname	mname	lname	users_id	
<input type="checkbox"/>		Редагувати	 Копіювати	 Видалити	1	Іван	Іванович	Кут	6
<input type="checkbox"/>		Редагувати	 Копіювати	 Видалити	3	Ольга	Олегівна	Дудар	7

Рисунок 3.9 – Результат додавання нового студента

Також протестовано форму додавання нового викладача (рис. 3.10). Заповнено її необхідною інформацією та збережено результат.

Галицький фаховий коледж

Групи Студенти Викладачі Предмети Оцінки

Викладачі + Створити Редагувати Знищити

	Прізвище	Ім'я	По-батькові
<input type="checkbox"/>	Кут	Іван	Іванович
<input type="checkbox"/>	Дудар	Ольга	Олегівна

Рисунок 3.10 – Форма додавання нового викладача

Інформація з цієї форми потрапляє в таблицю `#_gradebook_teachers` (рис. 3.11). Відкрити цю таблицю можна за допомогою phpMyAdmin, щоб впевнитися в тому, що запис даних відбувся.

+ Параметри

		id_teach	ffname	mname	lname	users_id
<input type="checkbox"/>	Редагувати Копіювати Видалити	1	Іван	Іванович	Кут	6
<input type="checkbox"/>	Редагувати Копіювати Видалити	3	Ольга	Олегівна	Дудар	7

Рисунок 3.11 – Результат додавання викладача

Наступним кроком тестування буде тестування форми додавання нового предмету (рис. 3.12). Заповнено її необхідною інформацією (назва предмету, викладач, навчальні групи, в яких він викладається) і збережено результат.

Галицький фаховий коледж

Групи Студенти Викладачі Предмети Оцінки

Предмети + Створити Редагувати Знищити

	Назва предмету	Викладач предмету	Навчальні групи
<input type="checkbox"/>	Математика	Кут Іван Іванович	КН-11
<input type="checkbox"/>	Біологія	Кут Іван Іванович	КН-21
<input type="checkbox"/>	Фізика	Дудар Ольга Олегівна	КІ-21
<input type="checkbox"/>	Історія України	Дудар Ольга Олегівна	КН-21

Рисунок 3.12 – Форма додавання нового предмету

Результат запису зберігається в таблицях #_gradebook_subjects (рис. 3.13) і #_gradebook_subj_to_group (рис. 3.14). В першій із них зберігається інформація про предмет, а друга використовується для зв'язування предметів і груп.

9	Предмет 1	8
11	ПІСК	9

Рисунок 3.13 – Інформація про доданий предмет в БД

	id_stg	group_id	subject_id
	64	31	10
	66	29	11
	65	1	11
	63	31	9
	54	1	4
	53	25	4
	52	29	4
	62	30	9

Рисунок 3.14 – Зв'язування нового предмету і груп в БД

Далі протестовано форму додавання оцінок (рис. 3.15).

Додати оцінку Зберегти Зберегти і закрити Скасувати

Предмет:

КН-11 - Математика (Кут Іван Іванович)
 КН-21 - Біологія (Кут Іван Іванович)
 КІ-21 - Фізика (Дудар Ольга Олегівна)
 КН-21 - Історія України (Дудар Ольга Олегівна)

Студент:

Козак Іван (КН-11)

Оцінка:

Рисунок 3.15 – Форма додавання оцінок

Інформація з цієї форми зберігається в таблиці #_gradebook_marks (рис. 3.16).

		id_mark	student_id	subject_id	mark
✖		1	16	4	5
✖		18	21	4	4
✖		17	20	4	4
✖		15	20	9	5

Рисунок 3.16 – Інформація про оцінку в БД


Наступним кроком тестування буде перевірка виведення інформації із бази даних. Для цього потрібно відкрити форми перегляду груп, студентів, викладачів, предметів і оцінок.

Результат тестування форми перегляду груп зображено на рисунку 3.17.

Назва групи	Звіт про групу
КН-11	Звіт про групу
КН-21	Звіт про групу
КІ-11	Звіт про групу

Рисунок 3.17 – Заповнена форма перегляду груп

Результат тестування форми перегляду студентів зображено на рисунку 3.18.


Галицький фаховий коледж

Групи
Студенти
Викладачі
Предмети
Оцінки

Студенти

+ Створити
Редагувати
Знищити

	Ім'я студента	Прізвище студента	Група
<input type="checkbox"/>	Іван	Козак	КН-11
<input type="checkbox"/>	Петро	Петренко	КН-21
<input type="checkbox"/>	Ольга	Хміль	КН-21
<input type="checkbox"/>	Марія	Марко	КІ-11
<input type="checkbox"/>	Андрій	Андрійов	КН-21

Рисунок 3.18 – Заповнена форма перегляду студентів

Результат тестування форми перегляду викладачів зображено на рисунку 3.19.

	Прізвище	Ім'я	По-батькові
<input type="checkbox"/>	Кут	Іван	Іванович
<input type="checkbox"/>	Дудар	Ольга	Олегівна

Рисунок 3.19 – Заповнена форма перегляду викладачів

Результат тестування форми перегляду предметів зображено на рисунку 3.20.

	Назва предмету	Викладач предмету	Навчальні групи
<input type="checkbox"/>	Математика	Кут Іван Іванович	КН-11
<input type="checkbox"/>	Біологія	Кут Іван Іванович	КН-21
<input type="checkbox"/>	Фізика	Дудар Ольга Олегівна	КІ-21
<input type="checkbox"/>	Історія України	Дудар Ольга Олегівна	КН-21

Рисунок 3.20 – Заповнена форма перегляду предметів

Результат тестування форми перегляду оцінок зображено на рисунку 3.21.



Оцінки

+ Створити

Редагувати

Знищити

	Група	Студент	Предмет	Оцінка
<input type="checkbox"/>	КН-11	Козак Іван	Математика	4
<input type="checkbox"/>	КН-11	Козак Іван	Фізика	5
<input type="checkbox"/>	КН-21	Петренко Петро	Математика	3
<input type="checkbox"/>	КН-21	Петренко Петро	Біологія	4
<input type="checkbox"/>	КН-21	Петренко Петро	Історія України	3
<input type="checkbox"/>	КН-21	Хміль Ольга	Фізика	5
<input type="checkbox"/>	КН-21	Хміль Ольга	Історія України	5
<input type="checkbox"/>	КН-21	Хміль Ольга	Біологія	4
<input type="checkbox"/>	КН-21	Андрійов Андрій	Англійська мова	3

Рисунок 3.21 – Заповнена форма перегляду оцінок

Далі проведено тестування генерації звітів. Для цього потрібно перейти на форму перегляду груп і перейти за посиланням “Звіт про групу” (рис. 3.22) для групи К-47.

Назва групи	Звіт про групу
КН-11	Звіт про групу
КН-21	Звіт про групу
КІ-11	Звіт про групу

Рисунок 3.22 – Посилання на генератор звітів

Після переходу за цим посиланням отримано сторінку з кнопкою друку та таблицею, в якій відображено усі оцінки студентів групи з усіх предметів, середній бал кожного студента та кількість оцінок. Загальний вигляд звіту відображено на рисунку 3.23.

Звіт							
Студент	Біологія	Історія України	Середній бал	Двійки	Трійки	Четвірки	П'ятірки
Петренко Петро	4	3	3.5	0	1	1	0
Хміль Ольга	4	5	4.5	0	0	1	1
Андрійов Андрій	2	2	2	2	0	0	0

[Повернутись назад](#)

Рисунок 3.23 – Звіт для групи

Для перевірки внесення оцінок із фронтальної частини сайту потрібно увійти в систему, як викладач, обрати предмет з якого потрібно виставити

оцінки, групу і заповнити форму, в якій будуть перелічені усі студенти (рис. 3.24).

Додати оцінку Зберегти Зберегти і закрити Скасувати

Предмет:

- КН-11 - Математика (Кут Іван Іванович)
- КН-21 - Біологія (Кут Іван Іванович)
- КІ-21 - Фізика (Дудар Ольга Олегівна)
- КН-21 - Історія України (Дудар Ольга Олегівна)

Студент:

Козак Іван (КН-11)

Оцінка:

Рисунок 3.24 – Форма внесення оцінок

Після натиснення кнопки “Надіслати” інформація з форми потрапляє в таблицю `#_gradebooks_marks`. Переглянути її, щоб переконатися в правильності роботи системи можна на рисунку 3.25).

+ Options

			id_mark	student_id	subject_id	mark
<input type="checkbox"/>			1	16	4	5
<input type="checkbox"/>			18	21	4	4
<input type="checkbox"/>			17	20	4	4
<input type="checkbox"/>			15	20	9	5

Check All / Uncheck All With selected:

Рисунок 3.25 – Інформація про оцінки в БД

Наступним кроком тестування є перевірка виведення оцінок в фронтальній частині. Для цього потрібно авторизуватись на сайті в ролі студента і перейти за посиланням для перегляду оцінок. Можна побачити таблицю з усіма предметами та оцінками для студента (рис. 3.26), логін якого використовується.

Ваші оцінки

Предмет	Оцінка
Системи штучного інтелекту	4
Предмет 1	5

Рисунок 3.26 – Оцінки студента

Під час проведення тестування було ретельно перевірено функціонування програмного продукту відповідно до поставлених вимог. Тестування охоплювало всі основні функції та можливості системи.

Усі передбачені функції системи працюють коректно та відповідають очікуваній поведінці. Введення даних, обробка запитів та виведення результатів виконуються без помилок. Взаємодія між різними модулями та компонентами системи відбувається належним чином.

Інтерфейс системи є зручним та інтуїтивно зрозумілим для користувачів. Навігація між різними розділами та функціями здійснюється без ускладнень. Візуальне представлення даних та елементів керування відповідає вимогам та очікуванням.

Система демонструє задовільну швидкість та час відгуку на запити користувачів. Навантаження на систему під час типових сценаріїв використання не призводить до суттєвого сповільнення або збоїв.

Загалом, проведене тестування підтвердило, що програмний продукт повністю відповідає поставленим вимогам та готовий до використання. Під час тестування не було виявлено критичних помилок або недоліків, які могли б завадити належному функціонуванню системи.

4 ТЕХНІКО-ЕКОНОМІЧНЕ ОБГРУНТУВАННЯ

4.1 Аналіз ринку

Розроблений програмний продукт є веб-застосунком, призначеним для автоматизації процесів обліку успішності студентів у навчальних закладах. Він інтегрується з системою керування контентом Joomla! та надає наступні можливості:

- для викладачів - вносити оцінки студентів в електронному вигляді, замість використання паперових відомостей;
- для студентів - переглядати свої поточні оцінки та історію успішності онлайн;
- для адміністраторів - формувати звіти про успішність випускних груп в зручному форматі.

Даний веб-застосунок є новим рішенням на ринку, оскільки він не має прямих аналогів. Існуючі системи керування навчанням (LMS) зазвичай є окремими веб-сайтами та не інтегруються з контентними системами навчальних закладів. Натомість, розроблений застосунок є компонентом Joomla!, що забезпечує тісну інтеграцію з основним сайтом закладу.

Потенційними замовниками цього продукту є навчальні заклади різних рівнів (школи, коледжі, університети), які використовують Joomla! для свого основного веб-сайту та потребують рішення для автоматизації обліку успішності студентів. Продукт може реалізовуватись як на локальному, так і на міжнародному ринку освітніх послуг.

Основними перевагами розробленого веб-застосунку є зручність використання для всіх учасників навчального процесу, інтеграція з існуючим веб-сайтом навчального закладу та можливість автоматизації процесів обліку успішності студентів.

Очікується стійкий попит на розроблене рішення серед навчальних закладів, оскільки воно дозволяє модернізувати та автоматизувати процеси

					КР. КН 24.552.04.000 ПЗ	Арк.
						47
Змн.	Арк.	№ докум.	Підпис	Дата		

обліку успішності студентів, підвищити їх ефективність та зручність для всіх учасників навчального процесу: студентів, викладачів, адміністрації. Перехід від паперових відомостей до електронного обліку оцінок зменшує витрати часу та ресурсів на ведення обліку.

Здобувачі освіти можуть легко відстежувати свої оцінки онлайн, викладачі мають зручний інструмент для внесення оцінок, а адміністрація отримує можливість формувати звіти в електронному вигляді.

Додатковим фактором, що сприяє попиту, є безкоштовність та відкритий вихідний код продукту, що знижує бар'єр входу для його впровадження в навчальних закладах та надає можливість вносити зміни й розширювати функціонал за потребою.

Основним методом продажу є безкоштовне поширення вихідного коду продукту через спеціалізовані репозиторії розширень Joomla!, що забезпечує широку доступність рішення для цільової аудиторії.

Підтримка та обслуговування можуть здійснюватись через онлайн-форуми спільноти розробників Joomla! на безкоштовній основі або на комерційній основі для окремих клієнтів за додаткову плату, що може включати розширену технічну підтримку, налаштування, інтеграцію та інші послуги.

Зробити точні прогнози обсягів продажу розробленого продукту складно через його безкоштовний характер. Однак, з огляду на широке поширення системи керування контентом Joomla! та її популярність серед освітніх установ, потенційна аудиторія користувачів є значною.

Основними конкурентами даного продукту є інші компоненти Joomla!, призначені для інтеграції систем керування навчанням (LMS), такі як Guru, Joomlalearn LMS Basic Edition, Joomdle та Joom-Reports. Проте ці рішення є повноцінними LMS з додатковим функціоналом, що створює додаткове навантаження на сервер та базу даних. Більшість з них є платними продуктами.

					<i>КР. КН 24.552.04.000 ПЗ</i>	Арк.
						48
Змн.	Арк.	№ докум.	Підпис	Дата		

Ключовими конкурентними перевагами розробленого продукту є:

- спеціалізований функціонал для обліку успішності без зайвих можливостей LMS, що оптимізує навантаження на сервер та базу даних;
- безкоштовність та відкритий вихідний код, що знижує бар'єр входу для впровадження;
- тісна інтеграція з системою Joomla! та простота встановлення як компонента;
- зручний та інтуїтивний інтерфейс для всіх категорій користувачів: студентів, викладачів та адміністраторів.

Завдяки цим ключовим перевагам розроблений продукт має потенціал для широкого поширення серед навчальних закладів, що використовують Joomla! для свого веб-сайту, незважаючи на наявність альтернативних рішень на ринку.

Життєвий цикл розробленого програмного продукту значною мірою визначатиметься двома ключовими факторами. По-перше, це активність спільноти розробників Joomla!. За умови, що спільнота буде активно підтримувати та оновлювати функціонал продукту відповідно до змін у ядрі Joomla!, він матиме тривалий життєвий цикл. Ви оцінюєте цей період у 5-10 років. Другим важливим фактором є попит на продукт серед навчальних закладів. Якщо попит з боку цільової аудиторії буде стабільним, це також сприятиме тривалому життєвому циклу.

Після 5-10 років експлуатації, ймовірно, знадобиться повна переробка та модернізація продукту з урахуванням нових технологічних тенденцій, що з'являться за цей час.

Загалом, аналіз ринку показує, що розроблений веб-застосунок має значний потенціал для комерціалізації та впровадження в освітніх установах завдяки кільком ключовим перевагам. По-перше, відсутність прямих аналогів на ринку. По-друге, вигідні цінові умови, а саме безкоштовність продукту. І, по-третє, тісна інтеграція з популярною системою керування контентом

					<i>КР. КН 24.552.04.000 ПЗ</i>	Арк.
						49
Змн.	Арк.	№ докум.	Підпис	Дата		

Joomla!, що полегшує впровадження та використання додатку в навчальних закладах, де вже використовується Joomla!.

Отже, за наявності належної підтримки та стабільного попиту, розроблений веб-застосунок має хороші перспективи для успішної комерціалізації та тривалого життєвого циклу.

4.2 Розрахунок витрат на проєктування

Розрахунок усіх витрат на проєктування веб-застосунку складається з одинадцяти пунктів. Таблицю кошторису зображено у додатку А.

Заробітна плата працівників визначається на основі кількості виконавців, діючих посадових окладів та кількості місяців їхньої участі в розробці проєкту.

Рекомендована кількість виконавців – до 3 осіб, тривалість розробки – 5 місяців. Розрахунок зарплати проводиться за формою таблиці 4.1.

Таблиця 4.1 – Розрахунок зарплати

N п/п	Посада працівника	Оклад, грн/міс	Відрахування, грн/міс	Кількість, чол.	Кількість, міс	З/п, грн
1	Інженер	10 510	2 312	1	5	63 610
2	Інженер II категорії	11 007	2 421	1	5	67 140
3	Технік	8 420	1 852	1	5	54 610
	Усього зарплати:					195 960

Посадові оклади визначаються шляхом множення ставки першого розряду, яка становить 2102 гривні, на тарифний коефіцієнт відповідної посади з подальшим округленням до цілого значення. Це означає, що для визначення окладу необхідно знайти тарифний коефіцієнт, який відповідає певній посаді, а потім помножити його на базову ставку 2102 гривні та округлити результат до найближчого цілого числа.

					КР. КН 24.552.04.000 ПЗ	Арк.
						50
Змн.	Арк.	№ докум.	Підпис	Дата		

Сума відрахувань на соціальні потреби розраховується шляхом застосування встановленого нормативу у відсотках від загальної суми заробітної плати. У наведеному прикладі цей норматив становить 22%. Таким чином, відрахування на соціальні потреби обчислюються як 22% від загальної суми заробітної плати 195 960 гривень, що дорівнює 43 112 гривням.

Контрагентські роботи – це роботи, які виконуються на замовлення організації-проектувальника іншими організаціями-субпідрядниками. Ціна таких робіт зазвичай становить від 10% до 20% від суми заробітної плати основних виконавців роботи. У наведеному прикладі вартість контрагентських робіт та послуг розраховується як 10% від суми заробітної плати 195 960 гривень, що дорівнює 19 596 гривням.

Витрати на відрядження зазвичай визначаються прямим підрахунком на основі передбачуваної кількості відряджень та їхньої середньої вартості. У цьому проекті заплановано 12 відряджень із середньою вартістю 1 000 гривень кожне. Таким чином, витрати на відрядження складають 12 000 гривень (12 відряджень * 1 000 гривень).

Інші прямі витрати включають вартість спеціального обладнання, витратних матеріалів та канцелярських товарів і можуть становити від 40% до 50% від видатків на заробітну плату. У наведеному прикладі інші прямі витрати розраховані як 50% від суми заробітної плати 195 960 гривень, що дорівнює 97 980 гривням.

Підсумкова сума прямих витрат записується як підсумок за пунктами 1-5, де:

- Заробітна плата - 195 960 гривень.
- Відрахування на соціальні потреби - 43 112 гривень.
- Контрагентські роботи і послуги - 19 596 гривень.
- Витрати на відрядження - 12 000 гривень.
- Інші прямі витрати - 97 980 гривень.

					КР. КН 24.552.04.000 ПЗ	Арк.
						51
Змн.	Арк.	№ докум.	Підпис	Дата		

Отже, загальна сума прямих витрат становить 368 648 гривень (195 960 + + 43 112 + 19 596 + 12 000 + 97 980).

Накладні витрати враховують загальногосподарські витрати, необхідні для забезпечення проведення роботи: витрати на опалення, електроенергію, амортизацію будівель, заробітну плату адміністративного персоналу тощо. Вони визначаються у відсотках (зазвичай від 30% до 40%) від суми прямих витрат за даною роботою. У наведеному прикладі накладні витрати розраховані як 30% від суми прямих витрат 368 648 гривень, що становить 110 594 гривні.

Планові накопичення – це прибуток, який спрямовується на розвиток матеріально-технічної бази організації-проектувальника та преміювання її працівників. Величина накопичень визначається у відсотках (зазвичай від 20% до 30%) від суми прямих і накладних витрат. У цьому випадку планові накопичення становлять 20% від суми прямих витрат (368 648 гривень) та накладних витрат (110 594 гривні), що дорівнює 95 308 гривням.

Сума прямих і накладних витрат, а також планових накопичень становить повну кошторисну вартість проєкту. Таким чином, кошторисна вартість проєкту розраховується як:

368 648 гривень (прямі витрати) + 110 594 гривні (накладні витрати) + + 95 308 гривень (планові накопичення) = 574 550 гривень.

Додатково до кошторисної вартості проєкту необхідно врахувати податок на додану вартість (ПДВ). ПДВ визначається за діючим нормативом (20%) від кошторисної вартості проєкту. У цьому випадку ПДВ становить:

574 550 гривень * 20% = 114 910 гривень.

Загальна договірна ціна розробки проєкту розраховується як сума кошторисної вартості робіт та податку на додану вартість. Вона визначає ті витрати на проєктування, які має покрити підприємство-замовник.

Отже, загальна договірна ціна розробки (Зп) дорівнює:

					КР. КН 24.552.04.000 ПЗ	Арк.
						52
Змн.	Арк.	№ докум.	Підпис	Дата		

574 550 гривень (кошторисна вартість) + 114 910 гривень (ПДВ) = 689 460 гривень.

4.3 Обґрунтування необхідності розробки

Сучасний навчальний процес вимагає ефективного управління та координації для забезпечення якісного надання освітніх послуг. Розробка веб-системи підтримки навчального процесу є необхідною для задоволення потреб різних зацікавлених сторін, таких як студенти, викладачі, адміністрація навчальних закладів, а також для підвищення ефективності та оптимізації навчального процесу.

Впровадження такої веб-системи дозволить задовольнити наступні потреби замовників (покупців):

1) Здобувачі освіти:

- зручний доступ до навчальних матеріалів, розкладів, оголошень у будь-який час та з будь-якого місця;
- можливість відстежувати свій прогрес, оцінки та виконувати завдання онлайн;
- ефективна комунікація з викладачами та одногрупниками.

2) Викладачі:

- централізоване управління навчальними ресурсами та матеріалами;
- можливість надавати завдання, перевіряти роботи та виставляти оцінки онлайн;
- зручний спосіб комунікації зі студентами та обміну інформацією.

3) Адміністрація навчальних закладів:

- ефективне управління навчальним процесом та розкладом;
- моніторинг успішності студентів та діяльності викладачів;
- оптимізація використання ресурсів та скорочення паперового документообігу.

					КР. КН 24.552.04.000 ПЗ	Арк.
						53
Змн.	Арк.	№ докум.	Підпис	Дата		

Впровадження веб-системи підтримки навчального процесу матиме позитивний вплив на низку економічних показників навчального закладу.

По-перше, система дозволить підвищити ефективність та продуктивність праці викладачів та адміністративного персоналу. Викладачам більше не потрібно буде витрачати час на заповнення паперових відомостей та журналів, оскільки всі оцінки вноситимуться в електронному вигляді. Адміністратори зможуть швидко формувати звіти про успішність та аналізувати дані без додаткових трудовитрат. Це підвищить загальну продуктивність праці.

По-друге, запровадження системи дозволить значно зменшити витрати на паперові носії та канцелярські витрати. Відмова від паперових журналів успішності, відомостей та інших документів дасть економію на придбанні канцтоварів.

Крім того, веб-система сприятиме оптимізації використання ресурсів та інфраструктури навчального закладу. Завдяки централізованому управлінню навчальним процесом та моніторингу успішності стане легше виявляти неефективне використання ресурсів та своєчасно його коригувати. Це дозволить раціональніше використовувати наявну інфраструктуру.

Отже, впровадження веб-системи для підтримки навчального процесу дозволить навчальному закладу оптимізувати витрати на персонал, канцтовари та інфраструктуру, що матиме позитивний вплив на загальні економічні показники діяльності.

Запровадження проекту з розробки веб-системи підтримки навчального процесу спрямоване на отримання ефекту за кількома ключовими напрямками:

По-перше, це підвищення якості надання освітніх послуг. Впровадження сучасної веб-системи покращить організацію навчального процесу, полегшить доступ до навчальних матеріалів та покращить комунікацію між учасниками, що в підсумку сприятиме підвищенню якості освіти.

					<i>КР. КН 24.552.04.000 ПЗ</i>	Арк.
						54
Змн.	Арк.	№ докум.	Підпис	Дата		

Другим важливим напрямком є покращення комунікації та обміну інформацією між учасниками навчального процесу - студентами, викладачами та адміністрацією навчального закладу. Веб-система забезпечить зручний спільний доступ до необхідних даних та ефективний інформаційний обмін.

Крім того, система дозволить оптимізувати управління навчальним процесом та ресурсами. Адміністрація матиме зручний інструмент для моніторингу успішності студентів, планування розкладу та раціонального розподілу ресурсів.

Для здобувачів освіти та викладачів система забезпечить зручний доступ до навчальних матеріалів та інструментів в онлайн-режимі, що значно спростить їхню роботу.

Нарешті, запровадження веб-системи дозволить скоротити витрати на паперовий документообіг та канцелярські потреби, перейшовши до електронного обліку успішності та обміну даними.

Таким чином, розробка веб-системи підтримки навчального процесу є необхідною для задоволення потреб замовників, підвищення ефективності навчального процесу та отримання економічних вигод для навчальних закладів.

					<i>КР. КН 24.552.04.000 ПЗ</i>	Арк.
						55
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

У процесі виконання роботи з використанням мови програмування PHP та програмного каркасу (фреймворку) Joomla було реалізовано веб-застосунок, який дозволяє висвітлювати оцінки студентів на веб-сайті Галицького фахового коледжу. Цей застосунок надає можливість викладачам вводити та оновлювати оцінки, а студентам — переглядати їх у режимі реального часу. Крім того, всі дані про оцінки зберігаються в базі даних Інтернет-ресурсу, що забезпечує надійне зберігання інформації та можливості доступ до неї в будь-який час.

Розроблена система надає кілька важливих можливостей. По-перше, вона дозволяє формувати списки студентських груп, що забезпечує зручне управління академічними даними. Адміністратор може легко додавати нові групи та оновлювати існуючі, що спрощує організацію навчального процесу.

Крім того, система підтримує закріплення викладачів за певними групами студентів. Це дозволяє забезпечити чітку відповідальність за навчальний процес і сприяє ефективному управлінню академічними програмами. Викладачі отримують можливість безпосередньо керувати своїми групами та контролювати навчальний процес.

Адміністратор системи має можливість вносити оцінки для здобувачів освіти, що є корисним у випадках, коли потрібно оперативно внести дані або виправити помилки. Разом з тим, викладачі також можуть самостійно вносити оцінки своїм студентам, що забезпечує оперативність оновлення інформації про академічні успіхи та дозволяє безпосередньо контролювати процес оцінювання.

Здобувачі освіти можуть переглядати свої оцінки в режимі реального часу. Це забезпечує прозорість процесу оцінювання та дозволяє студентам слідкувати за своїми академічними досягненнями, що сприяє їхній мотивації та покращенню навчальних результатів.

					<i>КР. КН 24.552.04.000 ПЗ</i>	Арк.
						56
Змн.	Арк.	№ докум.	Підпис	Дата		

Також система має функцію формування звітності про успішність груп. Це дозволяє адміністраторам та викладачам аналізувати академічні результати студентів, відстежувати прогрес кожної групи та приймати обґрунтовані рішення щодо навчального процесу. Звіти допомагають визначити слабкі та сильні сторони навчання, що є важливим для підвищення якості освіти.

При реалізації застосунку було використано шаблон програмування MVC (Model-View-Controller), що дозволяє розділити роботу з базами даних, логіку додатку і виведення інформації. Це спрощує розробку та підтримку продукту, дозволяючи легко його модифікувати та розширювати в майбутньому.

Ще однією з використаних технологій була технологія AJAX (Asynchronous JavaScript and XML). AJAX дозволяє взаємодіяти з сервером без перезавантаження всієї сторінки, що забезпечує більш динамічний користувацький досвід, збільшує швидкість сторінки і зменшує навантаження на сервер.

Перевагами даного застосунку є безкоштовність, відкритий код, невелике навантаження на сервер, зручність у користуванні та спрощене формування звітів. Він доступний для широкого кола користувачів і освітніх закладів, а відкритий код дозволяє адаптувати додаток до специфічних потреб.

Можливими шляхами вдосконалення є додавання різноманітних шкал оцінювання та збільшення кількості статистичної інформації, що надається. Це дозволить враховувати специфіку різних навчальних дисциплін та отримувати більш детальні звіти про успішність.

					<i>КР. КН 24.552.04.000 ПЗ</i>	Арк.
						57
Змн.	Арк.	№ докум.	Підпис	Дата		

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Довідник по CSS властивостям. Український веб-довідник. URL: <https://css.in.ua/css/properties> (дата звернення: 27.03.2024).
2. Довідник по HTML тегам. Український веб-довідник. URL: <https://css.in.ua/html/tags> (дата звернення: 23.03.2024).
3. Довідник по JavaScript. Український веб-довідник. URL: <https://css.in.ua/js/objects> (дата звернення: 04.03.2024).
4. Dexter Mark. Joomla! Programming - Crawfordsville: Addisson-Wesley, 2012. – 556 с.
5. Heggen G. Joomla! Development. A Beginner's Guide. – Crawfordsville: Addisson-Wesley, 2011. – 185 с.
6. Фрімен Е., Робсон Е. Head First. Програмування на JavaScript. Харків : Фабула, 2022. 671 с.
7. CSS підручник. W3Schools українською. URL: <https://w3schoolsua.github.io/css/index.html#gsc.tab=0> (дата звернення: 19.03.2024).
8. HTML підручник. W3Schools українською. URL: <https://w3schoolsua.github.io/html/index.html#gsc.tab=0> (дата звернення: 04.03.2024).
9. PHP підручник. W3Schools українською. URL: <https://w3schoolsua.github.io/php/index.html#gsc.tab=0> (дата звернення: 13.03.2024).
10. Ukrainian W. JavaScript Підручник. Основи вебпрограмування. W3Schools українською. URL: <https://w3schoolsua.github.io/js/index.html#gsc.tab=0> (дата звернення: 08.03.2024).

					<i>КР. КН 24.552.04.000 ПЗ</i>	Арк.
						58
Змн.	Арк.	№ докум.	Підпис	Дата		

ДОДАТКИ

Додаток А

Кошторис витрат на проектування

Найменування статей витрат	Сума, грн	Обґрунтування
1. Зарплата проєктувальників	195 960	
2. Відрахування на соціальні потреби	43 112	
3. Контрагентські роботи і послуги	19 596	
4. Витрати на відрядження	12 000	
5. Інші прямі витрати	97 980	
6. Усього прямих витрат	368 648	
7. Накладні витрати	110 594	
8. Планові накопичення	95 308	
9. Усього кошторисна вартість проєкту	574 550	
10. Податок на додану вартість	114 910	
11. Загалом договірна ціна розробки Зп	689 460	

					КР. КН 24.552.04.000 ПЗ	Арк.
						59
Змн.	Арк.	№ докум.	Підпис	Дата		

Додаток Б Діаграма варіантів використання

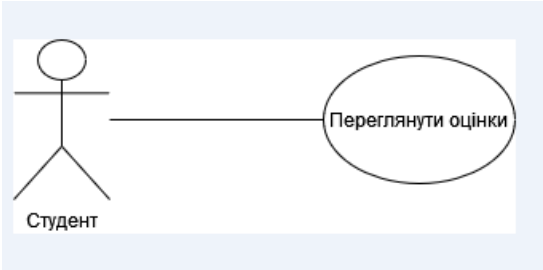
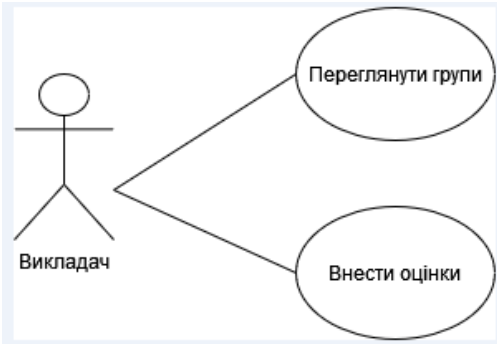
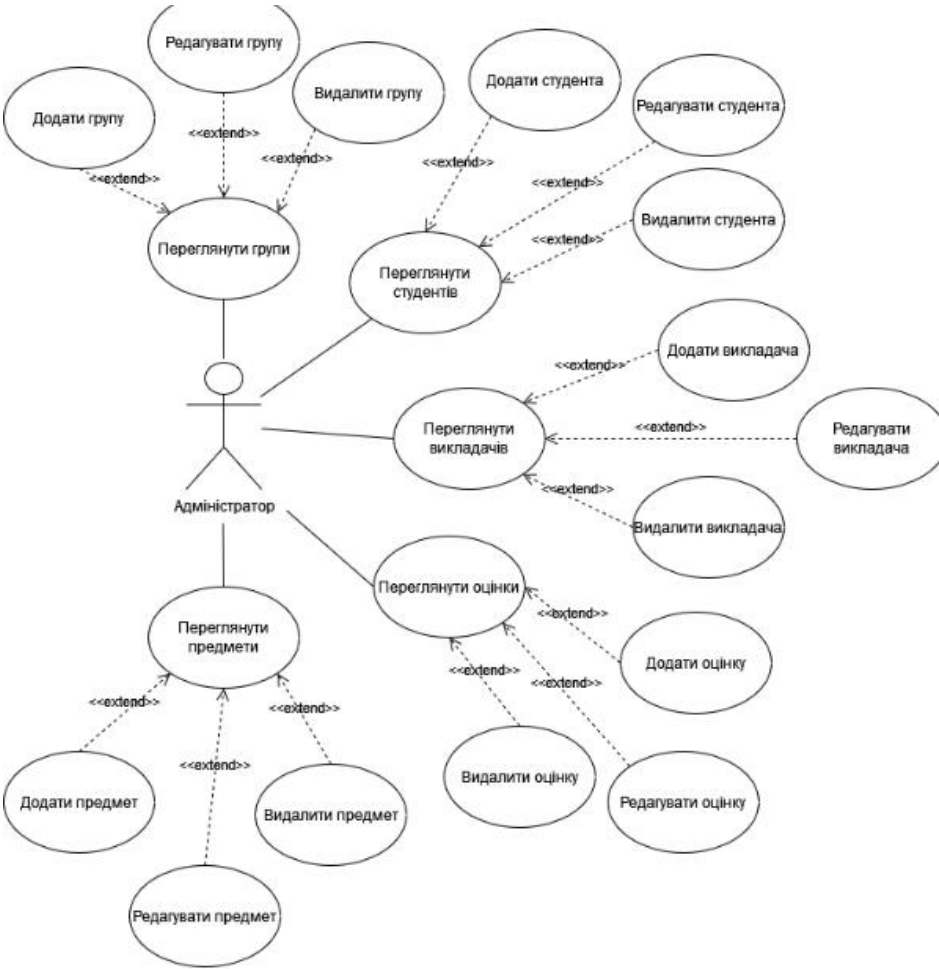


Рисунок 1.Б – Діаграма варіантів використання