

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»
Факультет інформатики та обчислювальної техніки
Кафедра інформаційних систем та технологій**

До захисту допущено:
Завідувач кафедри
_____ Олександр РОЛІК
«__» _____ 20__ р.

**Дипломний проєкт
на здобуття ступеня бакалавра
за освітньо-професійною програмою «Інформаційне забезпечення
робототехнічних систем»
спеціальності 126 «Інформаційні системи та технології»
на тему: «Онлайн-платформа для аналізу успішності студентів»**

Виконав (-ла):
студент (-ка) IV курсу, групи ІК-93
Нужний Олександр Юрійович _____

Керівник:
Викладач, кандидат наук, доцент,
Ткач Михайло Мартинович _____

Рецензент:
Доцент каф. ІП, к.т.н.,
Лісовиченко Олег Іванович _____

Засвідчую, що у цьому дипломному
проєкті немає запозичень з праць інших
авторів без відповідних посилань.
Студент (-ка) _____

Київ – 2023 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра інформаційних систем та технологій

Рівень вищої освіти – перший (бакалаврський)

Спеціальність – 126 «Інформаційні системи та технології»

Освітньо-професійна програма «Інформаційне забезпечення робототехнічних систем»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Олександр РОЛІК

«___» _____ 20__ р.

ЗАВДАННЯ

на дипломний проєкт студенту

Нужному Олександру Юрійовичу

1. Тема проєкту «Онлайн-платформа для аналізу успішності студентів», керівник проєкту Ткач Михайло Мартинович, к. т. н. затверджені наказом по університету від 31 травня 2023 №2101-с.

2. Термін подання студентом проєкту «12» червня 2023 року;

3. Вихідні дані до проєкту: Технічне завдання

4. Зміст пояснювальної записки:

1. Загальні положення: основні визначення та терміни, опис предметного середовища, огляд наявних аналогів, постановка задачі;

2. Інформаційне забезпечення: вхідні та вихідні дані, опис структури бази даних;

3. Програмне та технічне забезпечення: засоби розробки, вимоги до технічного забезпечення, архітектура програмного забезпечення, побудова звітів;

4. Технологічний розділ: керівництво користувача, методика випробувань

програмного продукту.

5. Перелік графічного матеріалу (із зазначенням обов'язкових креслеників, плакатів, презентацій тощо): алгоритм виставлення оцінок, алгоритм повідомлення новин, алгоритм здачі робіт, алгоритм створення курсу.

6. Дата видачі завдання: «9» лютого 2023 р.

Календарний план

№ з/п	Назва етапів виконання дослідницького проєкту	Термін виконання етапів проєкту	Примітка
1	Ознайомлення та аналіз потреб та вимог користувачів та бізнес-середовища, на основі чого проектування архітектури для веб-платформи, включаючи базу даних та взаємодію з користувачами.	12.05.2022	Виконано
2	Проектування, архітектури системи та визначення компонентів системи.	18.05.2022	Виконано
3	Розробка користувацького інтерфейсу, алгоритмів та методів потрібних для платформи	23.05.2022	Виконано
4	Реалізація функціоналу, тестування та оцінка продуктивності після чого вдосконалення та оптимізація платформи на основі отриманих результатів та фідбеку користувачів	03.06.2022	Виконано
5	Оформлення	08.06.2022	Виконано
6	Захист		

Студент

Олександр НУЖНИЙ

Керівник

Михайло ТКАЧ

АНОТАЦІЯ

Пояснювальна записка до дипломного проєкту: 63 сторінок, 12 рисунків, 14 таблиць, 1 додаток, 28 джерел.

Об'єктом дослідження є онлайн-платформа для аналізу успішності студентів.

Метою дипломного проєкту є розробка онлайн-платформи для аналізу успішності студентів.

Конкурентоспроможність будь-якого вищого навчального закладу залежить від його здатності і вміння готувати кваліфікованих фахівців, якість яких не тільки задовольняє вимогам споживачів і всіх зацікавлених сторін, але і перевершує їх очікування.

При перевірці знань, умінь і навичок велике значення має їх об'єктивна та незаангажована оцінка до якої висуваються наступні вимоги: оцінка повинна бути об'єктивною і справедливою, ясною і зрозумілою для студента, оцінка повинна виконувати стимулюючу функцію та орієнтувати на навчання, оцінка повинна бути всебічною та враховувати усі види навчальної діяльності.

JAVA SCRIPT, HYPERTEXT MARKUP LANGUAGE, VISUAL STUDIO CODE

SUMMARY

Explanatory note to the diploma project: 63 pages, 12 figures, 14 tables, 1 appendix, 28 sources.

The object of the study is an online platform for analyzing student performance.

The goal of the diploma project is the development of an online platform for analyzing the success of students.

The competitiveness of any higher education institution depends on its ability and ability to train qualified specialists whose quality not only meets the requirements of consumers and all interested parties, but also exceeds their expectations.

When checking knowledge, abilities and skills, their objective and non-involved evaluation is of great importance, to which the following requirements are made: the evaluation must be objective and fair, clear and understandable for the student, the evaluation must perform a stimulating function and focus on learning, assessment should be comprehensive and take into account all types of educational activities

JAVA SCRIPT, HYPERTEXT MARKUP LANGUAGE, VISUAL STUDIO
CODE

Номер рядка	Формат	Позначення	Найменування	Кільк. аркушів	Номер екзем.	Примітка
1			<u>Документація загальна</u>			
2						
3			Знову розроблена			
4						
5	A4	IK93.090БАК.005 ПЗ	Пояснювальна записка	63		
6	A4	IK93.090БАК.005 Д5	Онлайн-платформа для	12		
7			аналізу успішності студентів.			
8			Текст програмного коду			
9	A3	IK93.090БАК.005 Д1	Онлайн-платформа для	1		
10			аналізу успішності студентів.			
11			Алгоритм задачі роботи			
12	A3	IK93.090БАК.005 Д2	Онлайн-платформа для	1		
13			аналізу успішності студентів.			
14			Алгоритм створення курсу			
15	A3	IK93.090БАК.005 Д3	Онлайн-платформа для	1		
16			аналізу успішності студентів.			
17			Алгоритм виставлення			
18	A3	IK93.090БАК.005 Д4	Онлайн-платформа для	1		
19			аналізу успішності студентів.			
20			Алгоритм повідомлення			
21			новин			
22						
23						
24						
25						
26						
27						
28						
ЗМ.	Аркуш	№ докум.	Підпис	Дата		
Розроб.		Нужний О.Ю.			Онлайн-платформа для аналізу успішності студентів. Відомість проєкту Аркуш Аркушів 1 1 КПІ ім. Ігоря Сікорського Група ІК-93	
Керівн.		Ткач М.М.				
Затв.						

Пояснювальна записка
до дипломного проєкту
на тему: «Онлайн-платформа для аналізу успішності
студентів»

Київ – 2023 року

ЗМІСТ

ПЕРЕЛІК ПРИЙНЯТИХ СКОРОЧЕНЬ	4
ВСТУП	5
1 ПОСТАНОВКА ЗАВДАНЬ ДОСЛІДЖЕННЯ ТА ВИЗНАЧЕННЯ ВИМОГ ДЛЯ ОНЛАЙН-ПЛАТФОРМИ ДЛЯ АНАЛІЗУ УСПІШНОСТІ СТУДЕНТІВ	7
1.1 Змістовний опис і аналіз предметної області	7
1.2 Огляд і аналіз існуючих варіантів розробки онлайн-платформи для аналізу успішності студентів	10
1.3 Специфікація вимог до системи	14
Висновки до 1 розділу	16
2 ПОСТАНОВКА ЗАВДАННЯ.....	17
2.1 Розроблення архітектури онлайн-платформи для аналізу успішності студентів	17
2.2 Постановка задачі розробки.....	21
2.3 Розробка засобів реалізації програмного продукту.....	23
2.3.1 Вибір мови програмування	23
2.3.2 Вибір системи управління базами даних.....	28
Висновки до 2 розділу	33
3 ПРОЄКТНІ ТА ТЕХНІЧНІ РІШЕННЯ	35
3.1 Проєктування БД.....	35
3.2 Концептуальне рішення.....	37
3.3 Тестування програмного забезпечення.....	48
Висновки до 3 розділу	51

					І К93.090БАК.005 ПЗ							
Зм.	Лист	№ докум.	Підпис		Онлайн-платформа для аналізу успішності студентів. Пояснювальна записка				Літ.	Арк.	Аркушів	
Розробив	Нужний О.Ю.								Т		2	62
Перевірив	Ткач М.М.								КПІ ім. Ігоря Сікорського Група ІК-93			

4 ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ОТРИМАНИХ РЕЗУЛЬТАТІВ	52
4.1 Оцінка адекватності розробленої онлайн-платформи.....	52
4.2 Аналіз ефективності розробленого онлайн-платформи.....	55
Висновки до 4 розділу	57
ВИСНОВКИ.....	58
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	61
ДОДАТКИ	63

					ІК93.090БАК.005 ПЗ	Арк
Зм.	Лист	№ докум.	Підпис	Дата		3

ПЕРЕЛІК ПРИЙНЯТИХ СКОРОЧЕНЬ

СУБД – СИСТЕМА УПРАВЛІННЯ БАЗИ ДАНИХ

PC – PERSONAL COMPUTER

JS – JAVA SCRIPT

ТП – ТЕХНІЧНА ПІДТРИМКА

ПЗ – ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ

VSC – VISUAL STUDIO CODE

HTML - HYPERTEXT MARKUP LANGUAGE

CSS - CASCADING STYLE SHEETS

					ІК93.090БАК.005 ПЗ	Арк
						4
Зм.	Лист	№ докум.	Підпис	Дата		

ВСТУП

Дослідження онлайн-платформ для аналізу успішності студентів є надзвичайно актуальним у сучасній освітній системі. Ці платформи надають можливість університетам, вищим навчальним закладам та навчальним програмам збирати та аналізувати дані про студентів з метою поліпшення процесу навчання та забезпечення вищої якості освіти.

Аналіз успішності студентів дозволяє виявити сильні та слабкі сторони навчальних програм і методик, щоб вчителі могли адаптувати свої підходи до потреб студентів. Це може сприяти покращенню результатів навчання та підвищенню мотивації студентів.

За допомогою аналізу даних, платформи можуть збирати інформацію про індивідуальні успіхи студентів, їхні пріоритети та потреби. Це дає змогу налаштувати навчальний процес на основі цих даних, забезпечуючи індивідуальну підтримку кожному студентові.

Збирання та аналіз даних про успішність студентів дозволяє виявляти загальні тренди та патерни, які можуть бути корисними для розуміння ефективності навчальних програм та вдосконалення їх.

Онлайн-платформи дозволяють вимірювати ефективність різних навчальних стратегій та методик. Це допомагає установам вищої освіти зробити обґрунтовані вибори щодо навчальних планів, ресурсів та методів навчання.

Дослідження успішності студентів на онлайн-платформах надає велику кількість даних, які можуть бути використані для поліпшення навчальних програм, розробки нових курсів та стратегій, а також для підтримки прийняття рішень на рівні установ вищої освіти.

Таким чином, проблема, розглянута в цьому дипломному проекті, є важливою та має широкі практичні застосування як у сучасному світі, так і у майбутньому.

					ІК93.090БАК.005 ПЗ	Арк
Зм.	Лист	№ докум.	Підпис	Дата		5

Метою даного проекту є розробка онлайн-платформи для аналізу успішності студентів. Для досягнення цієї мети необхідно вирішити основні завдання:

- Аналіз платформ та сайтів для оцінки успішності навчання.
- Уточнення вимог до розробки інформаційної системи, яка дозволить користувачам зручно використовувати платформу.
- Розробка алгоритму, інтерфейсу, бази даних та реалізація програми

Відповідно до проведеного аналізу, перед системою поставлені такі основні функціональні задачі:

- Прискорення розвитку більш інноваційних систем для користувачів.
- Зниження часу, який користувачі витрачають на аналіз даних.
- Підвищення ефективності ведення обліку та управління навчальним процесом.

Загалом, дослідження онлайн-платформ для аналізу успішності студентів є важливим, оскільки вони можуть сприяти поліпшенню навчання, забезпечити індивідуальну підтримку студентів та допомогти встановленню ефективних навчальних стратегій.

					ІК93.090БАК.005 ПЗ	Арк
Зм.	Лист	№ докум.	Підпис	Дата		6

1 ПОСТАНОВКА ЗАВДАНЬ ДОСЛІДЖЕННЯ ТА ВИЗНАЧЕННЯ ВИМОГ ДЛЯ ОНЛАЙН-ПЛАТФОРМИ ДЛЯ АНАЛІЗУ УСПІШНОСТІ СТУДЕНТІВ

1.1 Змістовний опис і аналіз предметної області

Онлайн-платформи для оцінювання успішності студентів стають все більш важливими у шкільному навчанні, особливо з переходом до онлайн-класів. Сучасні педагоги використовують різні практики, щоб поліпшити академічну успішність та розвивати навички, впевненість і креативність учнів.

Використання онлайн-інструментів оцінювання дозволяє проводити комплексну перевірку успішності кожного учня і виявляти будь-які відмінності у їх навчанні. Традиційні методи оцінювання та відстеження даних можуть бути складними для використання у випадку великої кількості учнів, тому онлайн-інструменти оцінювання стають цінним ресурсом для вчителів, допомагаючи їм досягати вищих результатів.

Завдяки підходящим онлайн-інструментам оцінювання викладачі можуть легко створювати оцінки, використовуючи різноманітні ресурси. Ці інструменти не лише підвищують ефективність процесу створення оцінок, але й надають вчителям і учням більшу гнучкість.

Педагоги можуть використовувати онлайн-інструменти оцінювання для проведення формальних оцінювань. Швидкий доступ до даних формативного оцінювання допомагає вдосконалювати навчання учнів. Залежно від платформи, використовуваної вчителем, можна спростити процес створення, подання та оцінювання завдань та проектів. Крім того, можливості хмарних сервісів дозволяють зберігати всі підсумкові та формативні оцінки в одному місці.

Отже, онлайн-інструменти оцінювання стають необхідними для сучасних педагогів, допомагаючи їм покращувати процес навчання та досягати кращих результатів учнів.

					ІК93.090БАК.005 ПЗ	Арк
						7
Зм.	Лист	№ докум.	Підпис	Дата		

Онлайн-інструменти оцінювання дозволяють швидко створювати практичні тести для студентів, що полегшує їхнє навчання та дозволяє їм оцінити свій прогрес у засвоєнні навчальних показників. Використання таких інструментів допомагає забезпечити точну оцінку навчання студентів, що є важливим для їхнього розвитку.

Комплексні онлайн-інструменти оцінювання спрощують процес створення запитань, дозволяючи викладачам співпрацювати віддалено під час їх розробки. Вони можуть обмінюватися записами та коментувати запитання та підказки перед тим, як вони будуть надіслані студентам або додані до офіційних банків запитань. Також такі інструменти дозволяють додавати навчальні ресурси та призначати бали, забезпечуючи більш функціональне оцінювання.

Завдяки онлайн-інструментам оцінювання можна отримати повний зворотний зв'язок щодо результатів іспитів, а також статистичні дані про час, який студенти витрачають на кожне запитання. Використання таких інструментів дозволяє збирати цінні дані про успішність студентів та ефективність іспитів, що допомагає вдосконалювати формат тестів у майбутньому.

Онлайн-інструменти оцінювання також допомагають забезпечити конфіденційність іспитового матеріалу, захищаючи його від розголошення без дозволу. Вони також дозволяють використовувати схожі запитання для різних іспитів, що сприяє різноманітності тестування.

Отже, використання онлайн-інструментів оцінювання дозволяє викладачам зручно та ефективно створювати тести, забезпечуючи точну оцінку та зворотний зв'язок для студентів.

Традиційні методи оцінювання та відстеження даних можуть бути складними для використання у випадку великої кількості учнів, тому онлайн-інструменти оцінювання стають цінним ресурсом для вчителів, допомагаючи їм досягати вищих результатів.

					ІК93.090БАК.005 ПЗ	Арк
Зм.	Лист	№ докум.	Підпис	Дата		8

Завдяки використанню онлайн-інструментів оцінювання, учні можуть мати однакові освітні цілі та рівень складності, але кожен іспит може містити різні питання з невеликими варіаціями. Це забезпечує конфіденційність іспитів та зменшує ймовірність обміну відповідями.

Іспити відіграють важливу роль у оцінюванні знань студентів, незалежно від того, чи це щомісячні тести чи періодичні оцінки. Оскільки навчальний матеріал змінюється, важливо мати можливість регулярно оцінювати студентів. З використанням онлайн-інструментів оцінювання педагоги можуть проводити необмежену кількість іспитів протягом року, що дозволяє адаптувати навчальну програму для кращого навчання студентів.

За допомогою онлайн-технологій вчителі можуть надавати персоналізоване навчання для підвищення ефективності та успішності учнів. Розуміння даних оцінювання та їх використання для покращення результатів навчання є ключовим аспектом освіти, а використання онлайн-інструментів спрощує цей процес. Крім того, використання навчальних досліджень разом із даними оцінювання для індивідуальної персоналізації навчання для всього курсу або аудиторії є складним без використання віртуальних інструментів. Вибір найкращої програми онлайн-оцінювання полягає у знаходженні рішення, яке забезпечує індивідуальне оцінювання разом з різноманітними навчальними ресурсами та стратегіями.

У минулому недостатній час був обмежуючим фактором для вчителів у проведенні глибокого аналізу успішності та результатів оцінювання студентів. Проте, нові інструменти дозволяють більш ефективно аналізувати дані учнів та вдосконалювати плани уроків.

					ІК93.090БАК.005 ПЗ	Арк
						9
Зм.	Лист	№ докум.	Підпис	Дата		

1.2 Огляд і аналіз існуючих варіантів розробки онлайн-платформи для аналізу успішності студентів

Moodle – Moodle – це повноцінна система керування навчанням, яка може запропонувати персоналізований навчальний досвід, а також ефективно відстежувати його. Модульна конструкція платформи повністю настроюється. Він пропонує підтримку ряду інструментів для створення вмісту та навіть дозволяє масово завантажувати курси. Крім того, він достатньо гнучкий, щоб його можна було налаштувати для створення конкретних звітів, які вам потрібно переглянути, щоб відстежувати прогрес студентів.

Багато вчителів і професіоналів віддають перевагу Moodle, оскільки він є відкритим кодом, має багато інструментів для спільної роботи, таких як чат, форум і обговорення, полегшує ефективне керування курсом і користувачами, а також забезпечує легку інтеграцію з програмами сторонніх розробників, а також з користувальницькими плагінами

Merwin's space – Merwin's space – це безкоштовний інструмент для співпраці, який дозволяє викладачам створювати онлайн-класи, запрошувати студентів і проводити обговорення в класі. Щоб вчителі та учні могли використовувати платформу для електронного навчання, їхні школи повинні спочатку зареєструватися в Merwin's space.

Це спрощує процес створення, розподілу та оцінювання завдань. Оцінюванням, організацією класу та адмініструванням можна керувати в його інтерфейсі.

Moodle – Moodle робить усе, на що здатний Merwin's space. Але навпаки не вірно. Moodle величезний, просторий і пропонує набагато більше можливостей, ніж Merwin's space. Просто тому, що Merwin's space офіційно не є LMS.

					ІК93.090БАК.005 ПЗ	Арк
						10
Зм.	Лист	№ докум.	Підпис	Дата		

Moodle трохи складний для розуміння через його незграбний інтерфейс. Але його можна покращити за допомогою широкого спектру безкоштовних або платних тем Moodle і форматів курсів на ваш вибір.

Вчителі можуть розробити всю навчальну програму з нуля, оскільки Moodle підтримує безліч інструментів для створення вмісту. Moodle також надає батькам доступ до оцінок їхніх підопічних. Однак це завдання потребує певних зусиль з боку вчителя чи адміністратора. Хоча облікові записи для батьків не пропонуються в Merwin's space, він регулярно інформує батьків або відповідальних опікунів про успіхи учнів.

Окрім цього, Moodle забезпечує багатомовну підтримку, відстеження відвідуваності, відеоконференції через BigBlueButton, велику кількість модулів гейміфікації, дозволяє гейміфікацію тестів і вікторин тощо. Щоб переконатися, що учні залишаються на шляху і не кидають, переглядаючи тенденції участі, подання, та інші дані мають сенс. Це покращує загальний досвід електронного навчання, значно сприяючи показникам утримання та успішності студентів.

Moodle має хороші результати, коли мова йде про аналітику та звітність електронного навчання. Вкладка «Звіти» на панелі «Керування курсом» надає викладачам доступ до інструментів, які можна використовувати для редагування дат і груп для заходів або ресурсів або створення звітів про діяльність курсу на різних рівнях.

Merwin's space. Що стосується Merwin's space, ви не можете розробити власну навчальну програму в Merwin's space, але налаштувати заняття та завдання набагато легше. Ви навіть можете повторно використовувати тести чи завдання для майбутніх класів, установлювати права доступу, експортувати оцінки в таблицю Google, ділитися навчальними ресурсами у формі відео, посилань і зображень зі студентами.

					ІК93.090БАК.005 ПЗ	Арк
						11
Зм.	Лист	№ докум.	Підпис	Дата		

За допомогою Merwin's space ви можете легко керувати класними та домашніми завданнями в одному простому потоці. У ньому є все «зверху вниз» в одному гнучкому потоці, щоб студенти та викладачі могли легко знайти та отримати доступ.

Як початківець, ви можете вибрати Merwin's space, оскільки він має менш круту криву навчання. Merwin's space – це хмарна платформа, яка інтегрує Google Apps for Education і допомагає вчителям координувати повсякденну навчальну діяльність.

З огляду на те, що майже кожна людина має обліковий запис Gmail, Merwin's space може бути гарним варіантом для тих, хто не хоче вкладати свій час, гроші та зусилля в налаштування спеціальної LMS за допомогою Moodle.

Однак, якщо вам потрібна більша функціональність, активніша діяльність і оцінювання, Moodle — ідеальний варіант. Він також працює в автономному режимі через додаток, на відміну від Merwin's space, який працює виключно в хмарі.

Moodle – модульний дизайн Moodle робить його дуже універсальною LMS. Його можна налаштувати відповідно до потреб навчання поза академічним контекстом. Moodle LMS можна налаштувати відповідно до вимог шкіл K-12, коледжів, університетів, а також компаній.

Moodle також пропонує окрему спеціальну LMS під назвою Moodle Workplace. Він оновлений і вдосконалений, щоб відповідати всім вашим корпоративним потребам.

Порада: безкоштовні теми, як-от Fordson і Moove, найкраще підходять для дітей, і якщо ви хочете обрати платний варіант, тема Edwiser RemUI може надати чудовий інтерфейс для роботи дітей і вчителів. Переконайтеся, що ви використовуєте більше інтерактивних дій, гейміфікацію, навчання на основі аудіо та відео замість навчання на основі тексту. Крім того, учні люблять навчатися за допомогою ігор, балів, значків, головоломок, кросвордів, пошуків скарбів і вбудованих відео.

					ІК93.090БАК.005 ПЗ	Арк
						12
Зм.	Лист	№ докум.	Підпис	Дата		

Merwin's space – Merwin's space насамперед обслуговує навчання в середніх і старших школах. Він доступний безкоштовно для шкіл і включений у G Suite for Education, що робить його дуже зручним варіантом для проведення онлайн-занять. Крім того, Merwin's space пропонує відносно просту структуру та інтерфейс. Тому він більше підходить для тих, кому бракує технічного розуміння використання високофункціональної LMS.

1.3 Специфікація вимог до системи

Метою створення специфікації вимог до системи є визначення функцій системи та її не функціональних вимог. Першим кроком створення специфікації є опис глосарію проекту, який відображений у таблиці 1.1.

Таблиця 1.1 – Глосарій проекту

Термін	Опис терміну
1. Основні поняття та категорії предметної області та проекту	
Журнал обліку успішності	Документ в який записують інформацію про слухачів, вид, дату проведення заняття та присутність студентів. Журнал також є основою моніторингу в умовах кредитно-модульної системи
Викладач	Користувач системи, який має права визначати характеристики заняття виставляти оцінки, отримувати звіти
Студент (слухач)	Особа, що відвідує заняття та отримує оцінку за демонстрацію знань.

Академічна група	Перелік слухачів, які відвідують певні заняття та отримують оцінки
Режим реєстрації	Режим роботи системи під час якого викладач створюю доні що до проведеного заняття (вид, тема, час, наявність слухачів та їх оцінки)
Бали оцінювання	Чисельний або інший символічний еквівалент за допомогою якого викладач оцінює знання студентів
Вид заняття	Визначається типом взаємодії викладача та студентів, може мати значення (лекція, практичне заняття, семінар, лабораторна робота)
Показники поточної успішності	Інтегрований показник балів по відношенню до максимального його значення
Звітний документ	Документ отриманий з даних занесених в систему оцінювання знань

2. Користувачі системи

Викладач

Студент

3. Вхідні та вихідні документи	
Перелік академічних груп	Перелік академічних груп, які будуть занесені в систему
Список академічної групи	Прізвища студентів записані в базу даних і прив'язаних до академічних груп
Список дисциплін	Перелік дисциплін, які викладаються в відповідних академічних групах
Журнал відвідування	Документ де занесена інформація присутності студента та тому, або іншому занятті
Відомість оцінок	Документ, що пов'язує список академічної групи з оцінками за певною дисципліною
Витяг успішності студента	Перелік оцінок з в вибраних дисциплін для певного студента

Висновки до 1 розділу

Дослідження онлайн-платформи для аналізу успішності студентів є дуже актуальним у сучасній освітній системі. Такі платформи надають університетам, вищим навчальним закладам та навчальним програмам можливість збирати та аналізувати дані про студентів з метою вдосконалення процесу навчання та забезпечення кращої якості освіти.

Аналіз успішності студентів дозволяє ідентифікувати сильні та слабкі сторони навчальних програм і методик, щоб вчителі могли адаптувати свої підходи до потреб студентів. Це може сприяти покращенню результатів навчання та залученню студентів.

За допомогою аналізу даних, платформи можуть збирати інформацію про індивідуальні успіхи студентів, їхні пріоритети та потреби.

Це дає можливість налаштувати навчальний процес на основі цих даних, забезпечуючи індивідуальну підтримку кожному студентові.

					ІК93.090БАК.005 ПЗ	Арк
Зм.	Лист	№ докум.	Підпис	Дата		16

2 ПОСТАНОВКА ЗАВДАННЯ

2.1 Розроблення архітектури онлайн-платформи для аналізу успішності студентів

Веб система моніторингу якості отриманих знань містить у собі двох акторів – користувача(учня) системи та адміністратора(викладача);

На рисунку 2.1 представлена діаграма прецедентів.

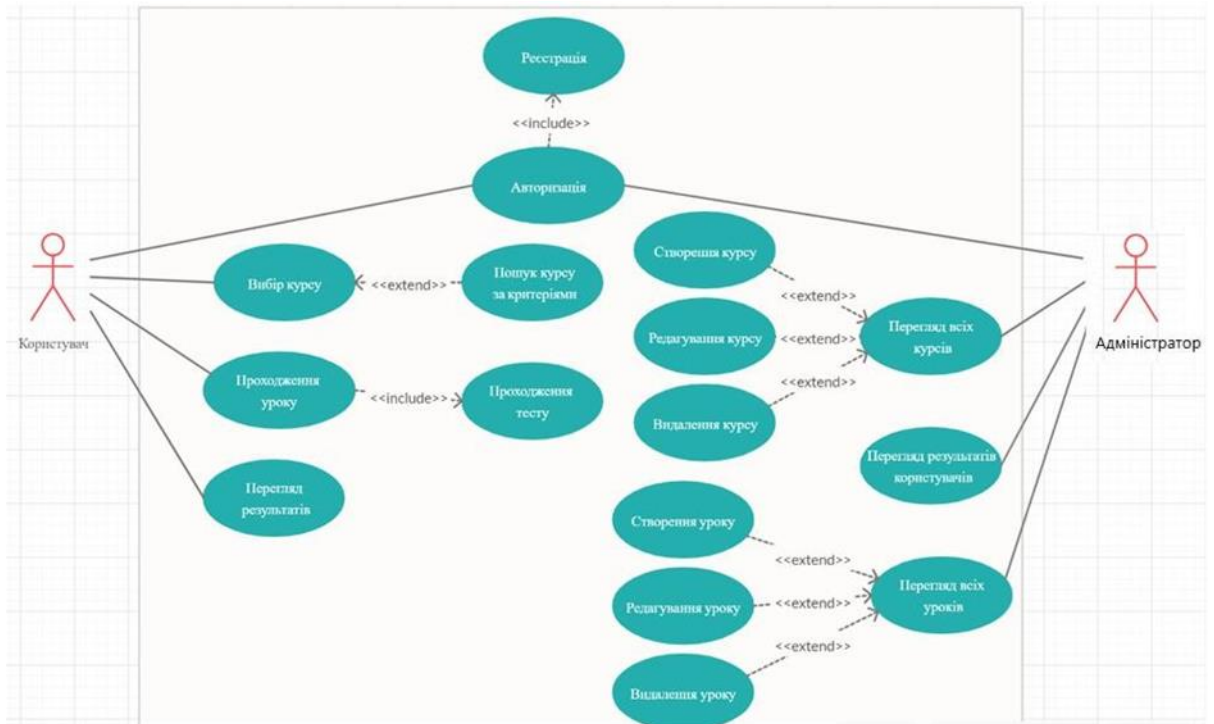


Рисунок 2.1 – Діаграма прецедентів

Для користувача «Студент» функції є значно обмежені і від має можливості тільки переглядати свої заборгованості та формувати відомість про свої заборгованості.

Далі розроблено табличний опис варіантів використання, які реалізують основну функцію.

Таблиця 1.1 – Варіант використання «Авторизація»

Контекст використання	Вхід в систему
Дійові особи	Користувач системи
Передумови	Початок роботи з системою
Триггер	Запуск програми на виконання
Сценарій	Ввести логін Ввести пароль Натиснути кнопку «Ok»
Постумова	Отримання доступу до елементів системи Стартове вікно «Панель керування»

Таблиця 1.2 – Варіант використання «Реєстрація заняття»

Контекст використання	Проведення заняття
Дійові особи	Користувач системи «Викладач»
Передумови	Вибрати пункт меню «Реєстрація заняття»
Триггер	Реєстрація заняття
Сценарій	Вибрати академічну групуВибрати дисципліну Ввести дату та місце проведення
Постумова	Відмітити присутність студентів

Таблиця 1.3 – Варіант використання «Відмітити відвідування»

Контекст використання	Внести інформацію про присутність студентів на конкретному занятті
Дійові особи	Користувач системи «Викладач»
Передумови	Виконати реєстрацію заняття
Триггер	Реєстрація заняття та встановлення присутніх на занятті
Сценарій	Отримати перелік студентів академічної групи Поставити відмітки про присутність (відсутність) Зберегти дані про переключку
Постумова	Система залишається в тому самому стані для внесення інших відміток (оцінка, коментар)

Таблиця 1.4 – Варіант використання «Виставити бали»

Контекст використання	Необхідність записати оцінку
Дійові особи	Користувач системи «Викладач»
Передумови	Виконати варіант використання «Реєстрація заняття»
Триггер	Реєстрація заняття та встановлення присутніх на занятті
Сценарій	Вибрати студента Записати в комірку на проти студента та під відповідним заняттям оцінку
Постумова	Система залишається в тому самому стані для внесення інших відміток (оцінка, коментар)

Таблиця 1.5 – Варіант використання «Перегляд успішності»

Контекст використання	Отримати інформацію про успішність
Дійові особи	Користувач системи «Викладач»
Передумови	Заповнені дані про академічні групи та виставлені оцінки
Триггер	Необхідність отримати інформацію про успішність студентів
Сценарій	Вибрати групу студентів Вибрати дисципліну Сформувати відомість успішності
Постумова	Повернення у вікно «Панель керування»

Таблиця 1.6 – Варіант використання «Перегляд заборгованості»

Контекст використання	Отримати інформацію про заборгованість
Дійові особи	Користувач системи «Викладач»
Передумови	Заповнені дані про академічні групи та відмітки про присутність
Триггер	Необхідність отримати інформацію про заборгованість студентів
Сценарій	Вибрати студента Вибрати дисципліну Сформувати відомість заборгованості
Постумова	Повернення у вікно «Панель керування»

2.2 Постановка задачі розробки

Системи дистанційного навчання призначені для навчання поза аудиторією. Метою використання цих систем є підвищення якості освіти, створення коротких курсів для підвищення якості студентів, а також накопичення та застосування знань у різних галузях науки, які вивчаються в різних навчальних закладах.

Реалізація має являти собою програмний продукт, який дозволяє створити середовище для вивчення нової інформації шляхом проходження курсів та забезпечення зворотного зв'язку щодо якості отриманих знань за результатами тестів та відеозаписів. З системою необхідно взаємодіяти через глобальну мережу Інтернет за допомогою протоколів прикладного рівня.

Користувачів системи слід розподілити відповідно до їхніх обов'язків і надати певні права доступу до функцій системи. Було вирішено створити два типи користувачів: викладач і студент.

Викладачі повинні мати доступ до можливості створювати інтерактивні уроки, а також додавати та редагувати інформаційні матеріали в системі. Повинна бути можливість контролювати успішність студентів.

Студенти матимуть доступ до курсів, курсів, складання іспитів, отримають відгуки та поради щодо результатів іспитів, а також матимуть можливість проконсультуватися з викладачами.

Система потребує створення механізмів збереження навчального матеріалу, а також можливості опису, редагування та пошуку. Усі доступні дані мають бути чітко структуровані та представлені у вигляді коефіцієнтів. Курс повинен включати уроки і завершуватися контрольною роботою студента.

Необхідно включати модулі зберігання інформації та підтримувати введення даних з інших баз даних, які зберігають дані, що поширюються в Інтернеті. Необхідно використовувати метамову, щоб продумати майбутній процес модернізації даних на наступних етапах.

					ІК93.090БАК.005 ПЗ	Арк
						21
Зм.	Лист	№ докум.	Підпис	Дата		

Потрібна безпека, оскільки система зберігає профілі студентів і записи. Паролі мають бути зашифрованими, а назва відеозаписів не повинна містити інформацію про користувача.

Доступ до освітніх ресурсів, створених викладачами, надається лише після реєстрації користувача. Після вивчення лекційного матеріалу студенти повинні виконати тестові завдання. Тести повинні містити запитання та варіанти відповідей. Після проходження тесту його результати та відеозапис користувача під час проходження курсу зберігаються в системі.

Тому вчитель може отримати доступ до результатів групи. Протягом курсу студенти повинні мати можливість консультуватися зі своїми викладачами. Підтримка редагування навчальних програм, системні навчальні програми є обов'язковим курсом.

Система дистанційного навчання повинна мати дружній і водночас інтуїтивно зрозумілий інтерфейс. Візуальний дизайн повинен розроблятися з урахуванням здібностей користувачів різного віку. Користувачі повинні з першого погляду зрозуміти, як користуватися системою.

Необхідно організувати модуль підтримки користувачів. Існує потреба у створенні платформи для системного довідкового та інформаційного розвитку з технологій електронного навчання в цілому.

Система дистанційного навчання має передбачати можливість модернізації та заміни модулів. Модель системи повинна забезпечувати її адаптацію до мінливих умов у процесі експлуатації та допускати поступову модернізацію окремих компонентів.

Щоб досягти цих цілей у процесі програмування, вам потрібно прийняти об'єктно-орієнтований стиль і дотримуватися вимог процесу програмування.

					ІК93.090БАК.005 ПЗ	Арк
						22
Зм.	Лист	№ докум.	Підпис	Дата		

2.3 Розробка засобів реалізації програмного продукту

2.3.1 Вибір мови програмування

В своєму дипломному проекті я обираю мову JavaScript, давайте розглянемо його специфіку та переваги.

JavaScript є найбільш широко використовуваною мовою в області розробки в усьому світі. Вже більше 25 років JavaScript Frameworks керує Інтернетом. Але код JavaScript обережно оброблявся в фреймворках JavaScript принаймні протягом останнього десятиліття.

JavaScript є однією з найпопулярніших мов і в даний час є вибором як для фронтальної, так і для серверної веб-розробки. Крім того, у середині фреймворку JavaScript ви, ймовірно, виберете досить скоро.

JavaScript Frameworks: Frameworks надає базову платформу для створення програм JavaScript для розробників. Це заощаджує розробникам роботу, щоб почати з функціональної основи, щоб почати все з нуля. Ця база включає в себе набір бібліотек коду в екземплярі JavaScript. Бібліотеки компілюють код для конкретного типу програми, з якою ви можете працювати. Фреймворк по суті сам визначить структуру програми. У кожній структурі JavaScript використовується інша функція. Для веб-розробки JavaScript є надійним вибором, і багато його структур базуються на цьому підприємстві.

JavaScript — це проста об'єктно-керована мова програмування, яка використовується для написання сценаріїв веб-сторінок на кількох веб-сайтах. Це інтерпретація, всеосяжна мова програмування. При застосуванні до HTML-документа JavaScript забезпечує динамічну інтерактивність веб-сайту.

JavaScript дає змогу користувачам створювати сучасні веб-застосунки без перезавантаження сторінки. JavaScript часто використовується для оновлення інтерфейсу користувача DOM API шляхом динамічної зміни HTML і CSS. Він широко використовується в онлайн-застосунках. Нижче наведено деякі варіанти використання мови програмування JavaScript.

					ІК93.090БАК.005 ПЗ	Арк
						23
Зм.	Лист	№ докум.	Підпис	Дата		

1. Веб-застосунки: будучи повсякденною розробкою браузерів, JavaScript стає все більш популярним для розробки потужних веб-застосунків. На прикладі Google Maps ми можемо це зрозуміти. Користувачеві потрібно лише клацнути та перетягнути мишею з Картами, деталі можна побачити одним клацанням миші. За цими поняттями використовується JavaScript.

2. Веб-розробка: для розробки веб-сторінок часто використовується JavaScript. Це дає нам змогу додати веб-сайту динаміку та надати конкретні функції веб-сайту. Він використовується в основному для перевірки на веб-сайтах. JavaScript дозволяє нам виконувати складні завдання і дозволяє користувачам взаємодіяти з веб-сайтами. Ви також можете завантажити вміст у документ за допомогою JavaScript, не завантажуючи веб-сторінку.

3. Мобільні програми. Мобільні телефони широко використовуються для доступу до Інтернету. Ми також можемо розробити невеб-контекстну програму з JavaScript. Це надійний інструмент для розробки мобільних застосунків за допомогою функцій і використання JavaScript. Фреймворк JavaScript часто використовується при створенні мобільних застосунків. За допомогою React Native ми можемо розробляти мобільні застосунки для кількох операційних систем. React Native виявився одним із найкращих фреймворків JavaScript для мобільних застосунків. Для операційних систем iOS і Android нам не потрібно писати різні коди. Нам просто потрібно створити та запустити його на кількох системах одночасно.

4. Ігри. Для створення ігор зазвичай використовується JavaScript. Він має різні бібліотеки та рамки для створення ігор. 2D або 3D гра. Механізми JavaScript, такі як PhysicsJS і Pixi.js, дозволяють нам створювати веб-ігри. WebGL, JavaScript API, також можна використовувати для відтворення 2D та 3D зображень у ваших браузерах. У нас є можливість запустити зображення.

					ІК93.090БАК.005 ПЗ	Арк
						24
Зм.	Лист	№ докум.	Підпис	Дата		

5. Презентації: Ми також можемо розробляти презентації як веб-сайт з JavaScript. Веб-коди слайдів можуть використовуватися такими бібліотеками, як RevealJ і BespokeJS. Ними легше керувати, тому за короткий час ми можемо легко зробити щось чудове. За допомогою HTML RevealJS створює інтерактивні та елегантні слайд-шоу. Ці лекції дуже добре працюють із мобільними пристроями та планшетами. Також підтримуються всі кольорові формати CSS.

6. Серверні програми: багато веб-програм мають серверну сторону. Для створення вмісту та обробки HTTP-запитів потрібен JavaScript. Ви також можете запустити JavaScript через NodeJS на серверах. NodeJS надає середовище з можливостями для запуску JavaScript на сервері.

7. Веб-сервери: Ви можете використовувати NodeJS для розробки веб-сервера. NodeJS керується подіями і не очікує відповіді на попередній виклик. Сервери NodeJS працюють швидко і не потребують буферизації або передачі фрагментів даних.

- CSS (Cascading Style Sheets) - це скоріше не мова програмування, а правила оформлення HTML документа. CSS дозволяє розробнику встановлювати розмітку сторінки, розмір шрифтів, колір, заголовки, а також змінювати шрифт. Можна сказати, що CSS безпосередньо відповідає за візуальну складову сайту, так як жоден HTML документ не може обійтися без CSS.

React — це фреймворк JavaScript з відкритим кодом, який насправді не є фреймворком. Але наразі це найпоширеніша технологія фронтенд розробки у світі. React, спочатку розроблений і досі підтримується Facebook і підтримується активною спільнотою з відкритим кодом, насправді є «бібліотекою» JS.

ReactJS відповідає за побудову ієрархії або відтворення компонентів інтерфейсу користувача та забезпечує підтримку як інтерфейсу, так і сервера.

Як і всі фреймворки та бібліотеки JavaScript, основним випадком використання React є створення динамічних інтерфейсів користувача, які оновлюються на стороні клієнта (користувача). Сторінку не потрібно оновлювати у браузері, вона оновлюється автоматично.

					ІК93.090БАК.005 ПЗ	Арк
						25
Зм.	Лист	№ докум.	Підпис	Дата		

Точно так, як це робить стрічка Facebook. Ось чому гігант соціальних мереж інвестував у розробку власної бібліотеки JavaScript, спеціально адаптованої до потреб його власних програм.

React зарекомендував себе як найбільш використовувана фреймворк/бібліотека фронтенду JS у світі, оскільки він популярний як серед розробників програмного забезпечення, так і серед спонсорів проектів. У чому полягає привабливість як для розробників програмного забезпечення та архітекторів, які створюють програми, так і для організацій, які інвестують у них?

Популярність ReactJS серед спільноти розробників ґрунтується на його гнучкості (що впливає з того факту, що це бібліотека, а не фреймворк – ми розглянемо цю відмінність більш детально пізніше), швидкості та відносно доступній кривій навчання.

Але існують інші бібліотеки та фреймворки JavaScript, які розробники «люблять» використовувати майже так само. У звіті про стан JavaScript за 2019 рік Svelte та Vue.js вважаються майже такими ж популярними (89%, 88% і 87% відповідно), що й React на основі рівня задоволеності розробників.

Станом на кінець жовтня 2020 року на популярному порталі вакансій Monster.com є 7678 відкритих вакансій для розробників/інженерів React – порівняно з 1029 для ролей Vue.js і лише 16 для Svelte. На StackOverflow.com, де публікуються лише доступні ролі розробників та IT-спеціалістів, наразі оголошено 520 вакансій React. Для розробників Vue.js і Svelte кількість вакансій становить 168 і 3 відповідно.

Насправді Monster містить список вакансій 7260 і StackOverflow 304 для інженерів Angular. Набагато більше, ніж для Vue.js або Svelte, що робить його, на деяку відстань, другою найбільш затребуваною серед роботодавців структурою/бібліотекою JS. Це незважаючи на те, що Angular займає лише 5 місце за рівнем задоволеності розробників.

					ІК93.090БАК.005 ПЗ	Арк
Зм.	Лист	№ докум.	Підпис	Дата		26

Як ІТ-аутсорсингова компанія, у K&C ми також можемо засвідчити той факт, що ми отримуємо значно більше запитів від організацій, які шукають підтримку React, ніж будь-яка інша альтернатива JavaScript.

Висновок полягає в тому, що, очевидно, існують різні фактори, які впливають на популярність певного фреймворку або бібліотеки JavaScript серед самих розробників і організацій, які наймають розробників. Однак React унікально популярний серед обох груп зацікавлених сторін.

Мінімальним кваліфікаційним критерієм є або має бути те, що даний фреймворк або бібліотека добре технічно відповідає специфікаціям програми. В ідеалі вибір буде гнучким для потреб широкого діапазону потенційних специфікацій. Це дозволить одній команді працювати з різними або всіма програмами роботодавця.

Задоволення розробника

Це також великий плюс, якщо розробники хочуть працювати з мовою кодування, фреймворком або бібліотекою. Це підвищить їхній рівень задоволеності роботою, що позитивно вплине на продуктивність і рівень утримання, що в кінцевому підсумку вплине на кінцевий результат.

Менший розрив у навичках

Але є й інші причини, чому React зараз є найпопулярнішим вибором для розробки інтерфейсу. Одна з них полягає в тому, що на ринку більше розробників React, ніж спеціалістів у будь-якій іншій доступній системі/бібліотеці JS. Це важливо, оскільки це робить наймання, яке ніколи не буває легким, коли йдеться про хороших інженерів програмного забезпечення, менш вузьким місцем.

Розумний спонсор проекту вирішить розвивати технології, максимально орієнтовані на майбутнє. Великий пул потенційних співробітників є одним із ключових факторів для перспективного проекту розробки програмного забезпечення. Якщо найняти нових членів команди для збільшення ресурсів або заміни тих, хто покидає роботу, особливо складно, це може виявитися серйозною проблемою, яка зрештою стане витратою – прямо чи опосередковано.

					ІК93.090БАК.005 ПЗ	Арк
						27
Зм.	Лист	№ докум.	Підпис	Дата		

Зарплати можуть стати завищеними або проекти затримуватися та/або поточна робота зірватися.

React є однією з найбільш перспективних технологій у розробці програмного забезпечення, оскільки існує сильний пул розробників React. Принаймні порівняно з іншими варіантами – існує загальна нестача кваліфікованих розробників програмного забезпечення порівняно з попитом. І більший попит на розробників React створює позитивне коло. Молодші розробники, які приходять на роботу, частіше вибирають спеціалізацію на технологіях, на які є більший попит.

Існують інші причини, чому React можна вважати особливо перспективною технологією. Бібліотека підтримується Facebook, однією з найбільших компаній у світі. Він також має особливо активну спільноту з відкритим кодом, частково завдяки високій кількості розробників React. Поєднання підтримки Facebook і великої та залученої спільноти з відкритим кодом означає, що React постійно оновлюється та вдосконалюється, і в осяжному майбутньому не існує ризику застаріти.

2.3.2 Вибір системи управління базами даних

Системи баз даних або СУБД — це програмне забезпечення, яке обслуговує збір електронних і цифрових записів для отримання корисної інформації та зберігання цієї інформації, відоме як системи баз даних/системи керування базами даних або СУБД. Метою стандартної бази даних є зберігання та пошук даних. Бази даних, такі як стандартні реляційні бази даних, спеціально розроблені для зберігання та обробки структурованих даних.

У середовищі бази даних доступ до даних здійснюється, змінюється, контролюється, а потім представляється в добре організованому вигляді, що дозволяє бізнес-корпораціям виконувати численні операції з обробки даних.

					ІК93.090БАК.005 ПЗ	Арк
						28
Зм.	Лист	№ докум.	Підпис	Дата		

Дані зазвичай організовуються у формі рядків і стовпців, щоб мінімізувати робоче навантаження та миттєво отримати точні результати.

Різні типи даних, які можна зберігати, обробляти або отримувати в системі керування базами даних, включають числові дані, часові ряди, текстові та двійкові дані.

Основні характеристики систем баз даних наведені нижче:

1) Менше дублювання

Database Systems надає певний ідентифікаційний номер для кожного запису. Маючи певний ідентифікаційний номер для всіх записів, користувачі не відчуватимуть помилок і проблем дублювання.

2) Обмежене резервування

Безсумнівно, існує висока ймовірність повторення даних, оскільки кілька користувачів використовують ту саму версію/програмне забезпечення бази даних для зберігання своїх файлів. Щоб уникнути великої ймовірності надмірності, СУБД пропонує єдине сховище даних і різні функції відображення даних.

3) Простота використання

Для використання СУБД не потрібно отримувати знатний досвід або технічні навички. Причина в тому, що всі ці інструменти містять гладкий і простий у використанні інтерфейс. Незалежно від того, чи знайомі ви з мовами програмування, ви можете легко використовувати запити для вставки, оновлення, видалення чи пошуку записів у системах баз даних.

4) Кілька макетів і презентацій

Database Systems має різні макети та формати презентацій, за допомогою яких можна легко вибрати знання та мовні параметри відповідно до його/її досвіду. Деякі бази даних містять параметри перекладу, які дозволяють переходити від одного макета до іншого, не змінюючи цілісність даних.

					ІК93.090БАК.005 ПЗ	Арк
						29
Зм.	Лист	№ докум.	Підпис	Дата		

5) Зменшує простір для зберігання

Державні та приватні компанії використовують системи баз даних для збереження величезної кількості даних, файлів, документів, медіа, аудіо та відео розширень. Компаніям потрібно багато місця для зберігання цих активів, але СУБД забезпечує належну інтеграцію, допомагаючи користувачам зменшити простір у порівнянні з традиційними системами. Ця функція також дозволяє підприємствам заощаджувати кошти.

6) Безпека даних

Безпека даних є першочерговою та важливою потребою для компаній, оскільки хакерство є поширеним явищем у цьому цифровому світі. СУБД доступна для всіх користувачів, співробітників, клієнтів, тому для обмеження кількох вікон необхідно застосовувати різні політики та правила.

Пам'ятаючи про це, системи баз даних побудовані з надійними функціями безпеки, які дозволяють компаніям захищати конфіденційну інформацію. Підприємства розгортають політики для обмеження доступу для окремих користувачів, що дозволяє їм мінімізувати порушення безпеки та атаки інсайдерів.

7) План відновлення та резервного копіювання даних

Сьогодні Database Systems пропонує варіанти відновлення та резервного копіювання даних. Компанії знають, що навмисні та ненавмисні події можуть статися будь-коли. Наприклад, у деяких випадках співробітники випадково видаляють дані, розробники видаляють або відкидають таблиці виробництва та виробництва.

Отже, СУБД вбудовано з опціями відновлення даних і планом резервного копіювання, щоб уникнути таких інцидентів. Вони працюють як постійний план зберігання, у якому неможливо видалити дані.

8) Підтримання цілісності

Системи баз даних містять схеми, параметри первинного та вторинного ключів, які дозволяють компаніям, особливо компаніям електронної комерції та магазинам інвентарю, підтримувати цілісність, послідовність і одночасність даних.

9) Покращена ефективність

За допомогою функцій та інструментів СУБД необроблена інформація перетворюється на цінну статистику. Компанії використовують цю статистику, щоб прийняти мудре та швидке рішення в середовищі реального часу. Це підвищує продуктивність бази даних і ефективність системи.

Щоб надати кожній таблиці доступ до інформації в базі даних, потрібно створити клас сутності на мові програмування Javascript, властивості якого будуть відображати поля таблиці в базі даних.

Репозиторій надає інтерфейс для доступу та маніпулювання даними, що зберігаються в базі даних.

Класи сутностей схожі на таблиці в базі даних, а поля моделі є відповідними копіями окремих значень рядків і стовпців.

Крім того, при застосуванні об'єктно-реляційних проекцій дані можна легко отримати з таблиць із зв'язками «багато до багатьох» і «один до багатьох».

Node.js та MongoDB є найкращими технологіями для реалізації проекту «Онлайн платформа для успішності студентів» з кількох ключових причин. По-перше, Node.js є платформою заснованою на JavaScript, яка дозволяє використовувати єдину мову програмування як на стороні клієнта, так і на сервері. Це спрощує розробку та підтримку коду, забезпечує швидкість розробки та зменшує затрати на навчання розробників.

Node.js також відомий своєю високою продуктивністю та масштабованістю. Він працює на одному потоці, що дозволяє ефективно обробляти багато запитів одночасно. Це особливо важливо для платформи, яка очікує великого обсягу запитів від студентів та викладачів.

					ІК93.090БАК.005 ПЗ	Арк
Зм.	Лист	№ докум.	Підпис	Дата		31

Крім того, велика кількість доступних модулів та бібліотек для Node.js дозволяє швидко розширювати функціональність платформи та використовувати перевірені рішення.

MongoDB, як NoSQL база даних, є ідеальним вибором для цього проекту. Вона забезпечує гнучкість та швидкість у роботі з даними, особливо коли маємо справу зі змінюваною структурою даних, як у випадку платформи для студентів. MongoDB також підтримує масштабованість та реплікацію, що дозволяє легко розширювати обсяг даних та забезпечувати високу доступність системи.

Крім того, Node.js та MongoDB хорошо підходять для розробки асинхронних та реал-тайм додатків. Це важливо для платформи, яка може включати функціональність спілкування між студентами та викладачами в режимі реального часу, таких як онлайн-чати або спільні робочі простори.

Узагалі, поєднання Node.js та MongoDB забезпечує швидку розробку, високу продуктивність, масштабованість та гнучкість для проекту «Онлайн платформа для успішності студентів». Вони сприятимуть стабільності, ефективності та надійності платформи, що важливо для її успішної реалізації та задоволення потреб користувачів.

Система «Онлайн платформа для успішності студентів» має задовольняти ряд вимог, які сприятимуть ефективному та зручному навчанню. Першою вимогою є можливість реєстрації та авторизації для студентів та викладачів, що забезпечить безпеку та відповідний рівень доступу до системи.

Друга вимога полягає у зручному управлінні профілями користувачів, де студенти та викладачі зможуть створювати та редагувати свої профілі, додавати особисту інформацію та налаштовувати облікові записи.

Третя вимога передбачає наявність курсів та навчальних матеріалів, які викладачі зможуть створювати та керувати, включаючи завдання, лекції, тести та інші навчальні ресурси. Студентам слід мати доступ до цих матеріалів та можливість взаємодіяти з ними.

Четвертою вимогою є можливість оцінювання та звітності, де викладачі зможуть оцінювати роботу студентів, виставляти оцінки та зберігати результати. Крім того, система повинна забезпечити доступність звітів та статистики щодо успішності студентів, що допоможе відстежувати прогрес та проводити аналіз результатів.

П'ятою вимогою є мобільна сумісність, де платформа повинна бути доступною на різних пристроях, зокрема на смартфонах та планшетах. Це забезпечить студентам та викладачам зручний доступ до навчального контенту та функціональності незалежно від їхнього місця перебування.

Нарешті, остання вимога полягає у забезпеченні високого рівня безпеки та захисту даних. Система повинна гарантувати захист від несанкціонованого доступу, шифрування даних, резервне копіювання та інші заходи безпеки, щоб забезпечити конфіденційність та цілісність інформації користувачів.

Враховуючи ці вимоги, використання Node.js та MongoDB в системі «Онлайн платформа для успішності студентів» є найкращим варіантом, оскільки ці технології забезпечують швидку розробку, високу продуктивність, масштабованість та гнучкість. Node.js дозволить побудувати ефективний та швидкий сервер, а MongoDB - надійну та гнучку базу даних, що важливо для успішного функціонування платформи та задоволення потреб користувачів.

Висновки до 2 розділу

Веб система моніторингу якості отриманих знань містить у собі двох акторів – користувача(учня) системи та адміністратора(викладача);

Для користувача «Студент» функції є значно обмежені і від має можливості тільки переглядати свої заборгованості та формувати відомість про свої заборгованості.

Найголовне – це сервер, за допомогою нього функціонує весь дадоток. Далі база даних за допомогою якої відбувається обмін даними.

Все інше це компоненти, плагіни, модулі за допомогою всього цього складається наш веб-додаток.

Системи дистанційного навчання призначені для навчання поза аудиторією. Метою використання цих систем є підвищення якості освіти, створення коротких курсів для підвищення якості студентів, а також накопичення та застосування знань у різних галузях науки, які вивчаються в різних навчальних закладах.

Користувачів системи слід розподілити відповідно до їхніх обов'язків і надати певні права доступу до функцій системи. Було вирішено створити два типи користувачів: викладач і студент.

Викладачі повинні мати доступ до можливості створювати інтерактивні уроки, а також додавати та редагувати інформаційні матеріали в системі. Повинна бути можливість контролювати успішність студентів.

Студенти матимуть доступ до курсів, курсів, складання іспитів, отримають відгуки та поради щодо результатів іспитів, а також матимуть можливість проконсультуватися з викладачами.

					ІК93.090БАК.005 ПЗ	Арк
		1				34
Зм.	Лист	№ докум.	Підпис	Дата		

3 ПРОЄКТНІ ТА ТЕХНІЧНІ РІШЕННЯ

3.1 Проєктування БД

Більш розширена інформація про структуру таблиць (ім'я, розмір і тип поля, описання поля) надана у таблицях 3.1 — 3.5.

Таблиця 3.1 — Таблиця «Користувач»

Ім'я поля	Тип	Опис поля
Id	bigint(20)	Первинний ключ
UserName	varchar(100)	Логін користувача
Email	varchar(100)	Адреса електронної пошти
PasswordHash	varchar(10000)	Хеш паролю
Premium	boolean()	Доступ до преміум курсів

Таблиця 3.2 — Структура таблиці «Курс»

Ім'я поля	Тип	Опис поля
Id	bigint(20)	Первинний ключ
Name	varchar(100)	Назва курсу
Description	varchar(500)	Опис курсу
Difficulty	varchar(10)	Складність
Type	boolean()	Тип (безкоштовний або преміум)

Таблиця 3.3 – Структура таблиці «Урок»

Ім'я поля	Тип	Опис поля
Id	bigint(20)	Первинний ключ
Name	nvarchar(100)	Назва уроку
Description	nvarchar(450)	Короткий опис
Text	mediumtext	Текст уроку
TestId	bigint(20)	Ключ тесту

Таблиця 3.4 – Структура таблиці «Тест»

Ім'я поля	Тип	Опис поля
Id	bigint(20)	Первинний ключ
Name	nvarchar(100)	Назва тесту
Question	nvarchar(500)	Питання
Answer	nvarchar(200)	Відповідь

Таблиця 3.5 – Структура таблиці «Результат»

Ім'я поля	Тип	Опис поля
Id	bigint(20)	Первинний ключ
Points	int()	Бали
Time	data()	Витрачений час
Video	nvarchar(100)	Посилання на відео
Result	nvarchar(500)	Результат
UserId	bigint(20)	Первинний ключ користувача

Технічні вимоги до комп'ютера, який відповідає мінімальним системним вимогам для розробки і розгортання створюваної програми, інсталяції інструментальних засобів розробки, що враховує особливості експлуатації програмного забезпечення, приведені для всіх використовуваних програмних засобів. У такому випадку для розробки сайту буде використовуватись Visual Studio Code. Також для розробки інформаційної системи будуть використанні такі мови програмування:

- Java Script
- MongoDB
- HTML
- CSS

Також, написання коду може бути на будь-якій операційній системі, наприклад:

- Windows 10
- Windows 7
- Windows 8
- Linux
- Unix

3.2 Концептуальне рішення

Розробка кабінету користувача

Кабінет користувача – є головним робочим місцем користувача у системі. Він містить у собі інші підмодулі, що виконують головні функції системи.

До даних модулів належать:

- модуль перегляду всіх доступних курсів;
- модуль проходження курсів/уроків;
- модуль тестів;
- модуль перегляду результатів

					ІК93.090БАК.005 ПЗ	Арк
Зм.	Лист	№ докум.	Підпис	Дата		37

Модуль кабінету викладача

Даний модуль дозволяє адміністратору:

- створювати, переглядати, редагувати курси та уроки;
- створювати тести;
- переглядати результати користувачів;

На основі теоретичних досліджень, викладених у першому розділі, та практичного досвіду взаємодії з українською освітньою системою (в умовах очного та дистанційного навчання) [18, 19] можна виділити взаємодію учасники процесу дистанційного навчання з достатньою вірогідністю правильності. Наступним чином – діяти в описаному порядку.

Основними учасниками взаємодії є Student (студент) і Teacher (викладач, викладач, лаборант тощо). Інші учасники: асистенти, куратори (студентські групи, мобільність студентів тощо), методисти, завідувачі кафедр/викладачі, заступники завідувачів профільних кафедр/викладачі (навчальна діяльність, педагогічна діяльність, виховна робота тощо).

Для організації дистанційного навчання кожен з учасників має власну послідовність дій, яка легко описується елементарними структурними блок-схемами.

Побудуємо блок-схеми алгоритму дій користувачів.

Алгоритм дії студента (рис. 3.1).

Першим етапом роботи студента є вибір предметів (курсів). Можливі дві ситуації: основні обов'язкові курси та предмети за вибором студента. У випадку базових обов'язкових предметів, предмет визначається кафедрою/факультетом/університетом і є обов'язковим для всіх студентів. Можна вибрати унікальні теми для викладачів і окремі курси (так звані підтеми або розділи курсу). У випадку курсів за вибором студенти можуть вибрати будь-яку з доступних пар «викладач-предмет», доступних у викладанні університету/коледжу/курсу. Для вибору курсу за вибором учні повинні ознайомитися з викладачем, структурою запропонованого курсу, тематикою лабораторних і практичних робіт, оцінюванням роботи тощо.

					ІК93.090БАК.005 ПЗ	Арк
						38
Зм.	Лист	№ докум.	Підпис	Дата		

У дистанційній освіті цей крок не є складним, необхідно лише отримати основну необхідну інформацію в Інтернеті (у першому випадку список обов'язкових курсів; у другому випадку інформацію про курси) – на сайтах університетів, факультетів, кафедри або будь-який інший інформаційний ресурс сайту

Взаємодії на цьому етапі дистанційного навчання з основних предметів: мінімум факультативних прямих або мінімум факультативних непрямих.

Взаємодія основних і додаткових предметів для дистанційного навчання на цьому етапі: мінімум факультативного прямого або мінімального факультативного опосередкованого (єдиним винятком є необхідність адміністративного погодження структури курсу між викладачами та керівництвом факультету, факультету, кафедри, але це поза межами цієї роботи).

Крок 2. Реєстрація на курс.

Для реєстрації на курси в системі Merwin's space студенти повинні мати так званий код доступу або посилання для запрошення. Студенти можуть приєднуватися до занять, використовуючи наявні засоби доступу.

Коди доступу чи посилання-запрошення можуть бути надіслані всім учням викладачем (або іншою відповідальною особою), або учні повинні самі знайти коди доступу (наприклад, з інших предметів навчального процесу).

Взаємодія цього етапу дистанційного навчання з основними предметами: мінімум факультативно непряме або середнє факультативно пряме.

Взаємодія основних і додаткових предметів для цього етапу дистанційного навчання: мінімальний факультативний непрямий або середній факультативний прямий.

Крок 3. Проходження лекційного (теоретичного) навчання.

Після запису на курс студенти мають доступ до навчальних матеріалів та лекцій з теорії. Це (зазвичай) включає: відвідування лекцій за допомогою відеоконференцзв'язку Meet (через посилання або код доступу), навчальні та структуровані матеріали (підручники, посібники, відеолекції, програми тощо) та зовнішні ресурси (публікації, Інтернет-джерела тощо).

					ІК93.090БАК.005 ПЗ	Арк
Зм.	Лист	№ докум.	Підпис	Дата		39

Лекції можуть мати обов'язкові елементи: відвідування, термін навчання тощо, або бути факультативними (виконуються на розсуд студента). Лекції можуть оцінюватися або не оцінюватися.

Взаємодії основних суб'єктів дистанційного навчання на цьому етапі: значні (або помірні) потрібні прямі або помірні (значні) необов'язкові непрямі.

Взаємодія основних і додаткових предметів для дистанційного навчання на цьому етапі: мінімум за вибором прямий або мінімум за вибором непрямої.

Крок 4. Проходження практичного\лабораторного навчання.

Після зарахування на курс студенти мають доступ до навчальних матеріалів практикуму/лабораторії та готуються до виконання практичних/лабораторних робіт. Як правило, практичні/лабораторні заняття включають: завдання для практичних/лабораторних робіт, доступ до практичних/лабораторних робіт за допомогою відеоконференції Meet (за посиланням або кодом доступу); додаткові матеріали (підручники, методики, відеолекції, схеми, інтернет-ресурси тощо).). Таке навчання може мати обов'язкові елементи: присутність, кінцеві терміни подання тощо; частково необов'язкові (наприклад, лише лабораторні/практичні роботи без конференцій) або повністю необов'язкові (деякі дисципліни можуть не мати практичних чи лабораторних робіт). Ця підготовка зазвичай оцінюється, але за бажанням вона може бути без оцінок.

Це етап дистанційного навчання взаємодії основних предметів: від середнього до максимального, від факультативного до критичного, прямого чи непрямого.

Взаємодія основних і додаткових предметів для дистанційного навчання на цьому етапі: мінімум за вибором прямий або мінімум за вибором непрямої.

Крок 5. Виконання поточних тестових, контрольних завдань.

Протягом курсу від студентів може знадобитися пройти безперервний моніторинг продуктивності у формі тестів, контролю та подібних завдань. Ці завдання можуть мати терміни, варіанти, визначений порядок виконання тощо, які встановлює вчитель. При виконанні таких завдань студенти зазвичай отримують бал за шкалою оцінювання.

					ІК93.090БАК.005 ПЗ	Арк
Зм.	Лист	№ докум.	Підпис	Дата		40

Ця крокова дистанція вивчає взаємодію ключових гравців: від помірних до важливих, від необов'язкових до критичних, прямих чи непрямих.

Взаємодія основних і додаткових предметів для дистанційного навчання на цьому етапі: мінімум за вибором прямий або мінімум за вибором непрямий.

Крок 6. Виконання остаточних (фінальних) тестових, контрольних завдань (іспити, заліки тощо).

Наприкінці курсу студенти, як правило, проходять підсумкові (підсумкові) контрольні завдання у формі заліків, контролю, оцінювання, перевірок тощо. Такі завдання повинні мати фіксовані дати, тривалість, варіанти, визначений порядок виконання тощо, що встановлюється викладачем і завчасно повідомляється студентам. Виконуючи ці завдання, студенти зазвичай отримують оцінку на основі шкали оцінювання, повідомленої студенту заздалегідь.

Для виконання таких завдань можна використовувати аудіовізуальні комунікації або текстові версії. Виконання та подача завдань може бути письмовим, усним або комбінованим.

Взаємодія основних суб'єктів дистанційного навчання на даному кроці: від помірної до максимальної, від опціональної до критичної, пряма або непряма.

Взаємодія основних та додаткових суб'єктів дистанційного навчання на даному кроці: мінімальна опціональна пряма або мінімальна опціональна непряма.

					ІК93.090БАК.005 ПЗ	Арк
						41
Зм.	Лист	№ докум.	Підпис	Дата		

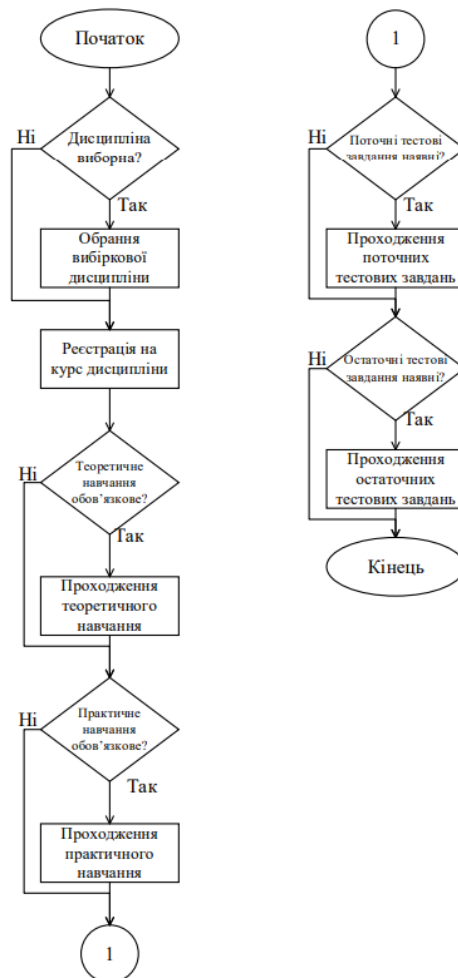


Рисунок 3.1 – Алгоритм дії студента

Алгоритм дії викладача (рис. 3.2).

Крок 1. Створення навчально-методичної бази дисципліни.

Викладач, який викладає даний курс, повинен спочатку створити навчально-методичні основи дисципліни. Це база для викладацької діяльності, що включає дистанційне та очне навчання. Однак інколи окремі моменти навчально-методичної бази необхідно змінити, щоб вони відповідали нормам і правилам дистанційного навчання.

При створенні навчально-методичних основ дисципліни викладачі включають до навчальної програми всю необхідну для виконання курсу інформацію та критерії оцінювання роботи. В авторській розробці можуть брати участь асистенти, методисти та інші додаткові об'єкти навчального процесу.

Взаємодія основних предметів дистанційного навчання на цьому етапі: мінімальний, факультативний.

Взаємодія основних і додаткових предметів для дистанційного навчання на цьому етапі: мінімум за вибором прямий або мінімум за вибором непрямої.

Крок 2. Створення «Класу» дисципліни.

Заняття створюються через особистий кабінет викладача в системі. Додаючи новий курс, ви повинні заповнити всі обов'язкові поля та необов'язкові (але рекомендовані) для заповнення необов'язкових полів.

На основі навчально-методичної бази 1 кроку вчитель доповнює всю необхідну інформацію. Тобто реалізує базу даних у вигляді електронного онлайн-курсу, через який відбуватиметься навчання.

Крім створення та наповнення курсів, можуть залучатися асистенти, методисти та інші види взаємодії з іншими суб'єктами навчального процесу.

Взаємодія основних предметів дистанційного навчання на цьому етапі: мінімальний, факультативний.

Взаємодія основних і додаткових предметів для дистанційного навчання на цьому етапі: мінімум за вибором прямий або мінімум за вибором непрямої.

Крок 3. Запрошення студентів

Для початку уроку вчителю потрібно викликати всіх учнів класу. Для всіх учнів з обов'язкових предметів - група\потік\курс. Для виробничих дисциплін - усі студенти, які виявили бажання вступати саме на цей курс.

Щоб зарахувати студентів на курс, викладачам необхідно створити електронний ресурс для цього курсу. В рамках системи – створити «клас». Класи створюються з «кодом класу» та посиланням для запрошення студентів. Цей код курсу або посилання для запрошення необхідно надати всім зацікавленим студентам. Цього можна досягти шляхом прямої взаємодії між студентами та викладачами, або шляхом публічного розміщення цієї інформації на інформаційних ресурсах університету/факультету/кафедри.

					ІК93.090БАК.005 ПЗ	Арк
Зм.	Лист	№ докум.	Підпис	Дата		43

Після того, як студент приєднався до класу, вчитель повинен перевірити список студентів на наявність усіх необхідних студентів, додаткових студентів тощо.

Взаємодія основних суб'єктів дистанційного навчання на даному кроці: мінімальна, критична, пряма або непряма.

Взаємодія основних та додаткових суб'єктів дистанційного навчання на даному кроці: мінімальна опціональна пряма або мінімальна опціональна непряма.

Крок 4. Проведення лекційного (теоретичного) навчання.

Лекційне (теоретичне) навчання зазвичай включає: лекції з використанням відеоконференцзв'язку Meet, використання навчального та структурованого матеріалу (обговорення зі студентами, вирішення проблем тощо) та аналіз зовнішніх освітніх ресурсів. Залежно від початкової методологічної основи дисципліни вона може мати обов'язкові або факультативні елементи. Лекції також можуть оцінюватися за встановленими критеріями чи ні.

Ця крокова дистанція вивчає взаємодію ключових гравців: від помірних до важливих, від необов'язкових до критичних, прямих чи непрямих.

Взаємодія основних і додаткових предметів для дистанційного навчання на цьому етапі: мінімум за вибором прямий або мінімум за вибором непрямий.

Крок 5. Проведення практичного (лабораторного) навчання.

Практична (лабораторна) підготовка студентів здійснюється на навчально-методичній основі і, як правило, включає: виконання практичних/лабораторних робіт, захист практичних/лабораторних робіт з використанням відеоконференцзв'язку Meet, додатковий матеріал. Така підготовка може бути обов'язковою, частково факультативною або повністю факультативною. Крім того, практична підготовка зазвичай оцінюється, але може не оцінюватися, якщо вона не є обов'язковою.

Це етап дистанційного навчання взаємодії основних предметів: від середнього до максимального, від факультативного до критичного, прямого чи непрямого.

Взаємодія основних і додаткових предметів для дистанційного навчання на цьому етапі: мінімум за вибором прямий або мінімум за вибором непрямої.

Крок 6. Проведення поточних тестових, контрольних завдань.

Під час уроків може проводитися поточний контроль для оцінки поточної успішності студента. Форма, предмет і тривалість цього контролю визначаються навчально-методичною базою, про що студенти повідомляються заздалегідь. При здачі завдань поточного контролю студентам, як правило, виставляється оцінка за попередньо оголошеною шкалою оцінювання.

Ця крокова дистанція вивчає взаємодію ключових гравців: від помірних до важливих, від необов'язкових до критичних, прямих чи непрямих.

Взаємодія основних і додаткових предметів для дистанційного навчання на цьому етапі: мінімум за вибором прямий або мінімум за вибором непрямої.

Крок 7. Проведення остаточних (фінальних) тестових, контрольних завдань (іспити, заліки тощо).

Наприкінці курсу студенти, як правило, повинні пройти певну визначену форму остаточної (підсумкової) перевірки успішності. Підсумковий контроль встановлюється навчально-методичною базою, обов'язково зазначаються дати, тривалість виконання, варіанти, порядок виконання тощо. Виконуючи ці завдання, студенти зазвичай отримують оцінку на основі шкали оцінювання, повідомленої студенту заздалегідь.

Остаточні оцінки можна зробити за допомогою інструменту спілкування онлайн-конференції або текстової версії. Виконання та подача завдань може бути письмовим, усним або комбінованим.

Це етап дистанційного навчання взаємодії основних предметів: від середнього до максимального, від факультативного до критичного, прямого чи непрямого.

					ІК93.090БАК.005 ПЗ	Арк
Зм.	Лист	№ докум.	Підпис	Дата		45

Взаємодія основних і додаткових предметів для дистанційного навчання на цьому етапі: мінімум за вибором прямий або мінімум за вибором непряий.

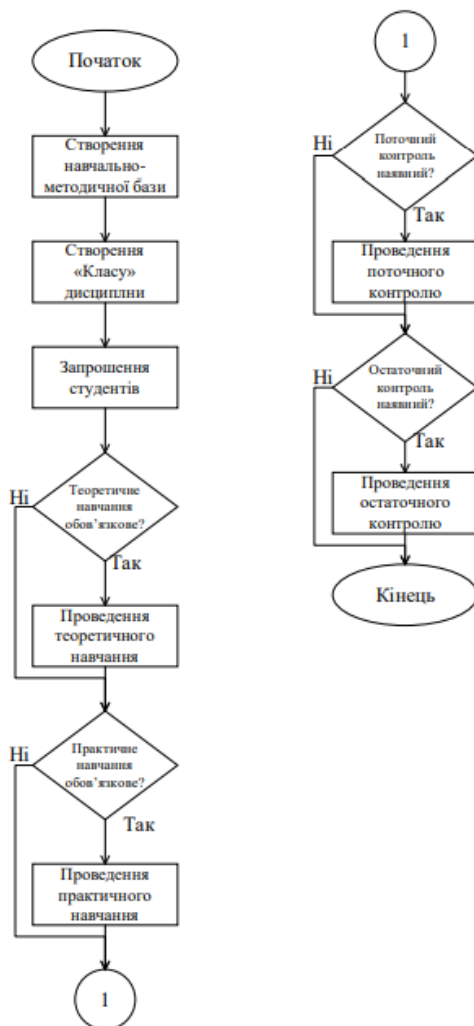


Рисунок 3.2 – Алгоритм дії викладача

Розглянемо для прикладу автоматизацію надсилення електронних листів всім студентам, які не здали роботу за день до терміну здачі (рис. 3.3).

Для навчально-методичних комплексів нагадування та модулі планування можуть бути розроблені за допомогою автоматизації електронної пошти та встановлення нагадувань для груп учнів у календарі.

Для автоматизації процесу нагадування та планування необхідно:

- 1) Отримати список студентів заданого курсу;
- 2) Отримати список завдань (практичних, лабораторних робіт тощо);
- 3) Отримати список термінів здачі завдань;
- 4) Отримати список студентів, які вже здали завдання.

Задані кроки реалізуються за допомогою API та згодом можуть бути поєднані в модуль автоматизації нагадувань із використанням, наприклад, зовнішніх засобів автоматизації електронної пошти або календаря.

В результаті отримується чотири масиви, які шляхом взаємного порівняння та об'єднання дозволяють налаштувати оптимальну автоматизацію нагадування та планування навчальних робіт шляхом розроблення повноцінних програмних застосунків (або інтеграції з існуючими засобами).

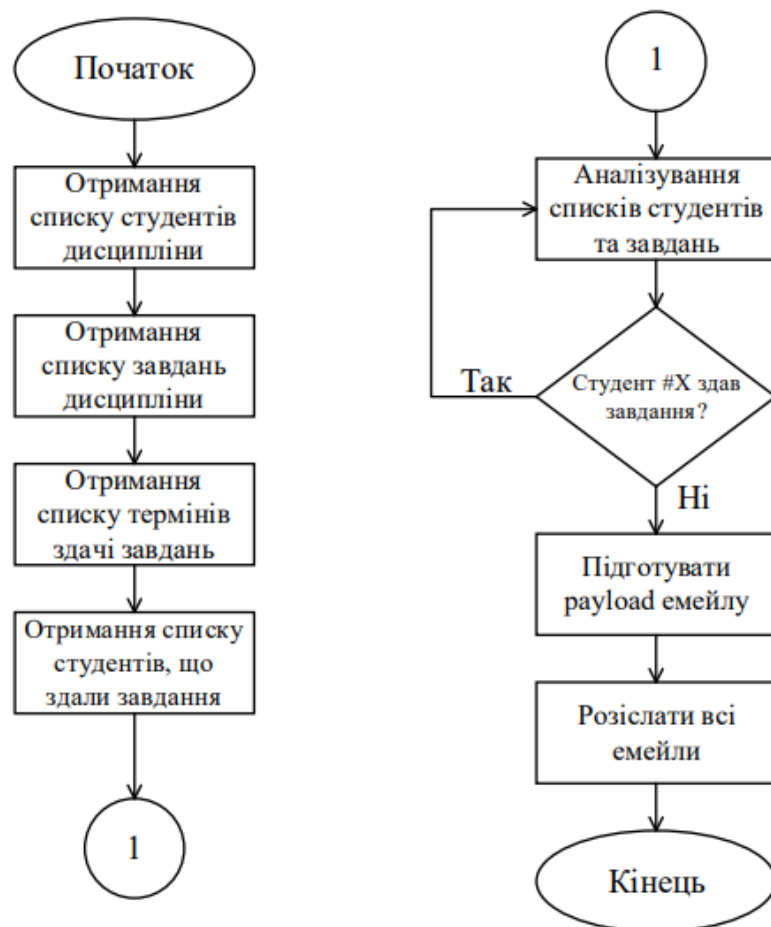


Рисунок 3.3 - Алгоритм нагадування за допомогою електронних листів

3.3 Тестування програмного забезпечення

Тестування програмного забезпечення – це процес технічного дослідження, призначений для виявлення інформації про якість продукту відносно контексту, в якому він має використовуватись. Техніка тестування також включає як процес пошуку помилок або інших дефектів, так і випробування програмних складових з метою оцінки. Може оцінюватись:

- відповідність вимогам, якими керувалися проектувальники та розробники;
- правильна відповідь для усіх можливих вхідних даних;
- виконання функцій за прийнятний час;
- практичність;
- сумісність з програмним забезпеченням та операційними системами;
- відповідність задачам замовника.

Тестування програмного забезпечення традиційно відокремлювалося від решти розробки. Його часто проводять пізніше в життєвому циклі розробки програмного забезпечення після етапу створення або виконання продукту. Тестер може мати лише невелике вікно для перевірки коду – іноді безпосередньо перед виходом програми на ринок. У разі виявлення дефектів часу на перекодування або повторне тестування може бути мало. Це не рідкість випускати програмне забезпечення вчасно, але з помилками та необхідними виправленнями. Або команда тестування може виправити помилки, але пропустити дату випуску.

Виконання тестових дій на початку циклу допомагає тримати тестування на передньому плані, а не залишатися позаду після розробки. Попередні тести програмного забезпечення також означають, що усунення дефектів обходиться дешевше.

Зараз багато команд розробників використовують методологію, відому як безперервне тестування. Це частина підходу DevOps, де розробка та операції співпрацюють протягом усього життєвого циклу продукту.

					ІК93.090БАК.005 ПЗ	Арк
						48
Зм.	Лист	№ докум.	Підпис	Дата		

Мета полягає в тому, щоб пришвидшити доставку програмного забезпечення, збалансувавши вартість, якість і ризик. Завдяки цій техніці тестування командам не потрібно чекати, поки буде створено програмне забезпечення, перш ніж почати тестування. Вони можуть проводити тести набагато раніше в циклі, щоб швидше виявити дефекти, коли їх легше виправити.

Оскільки число можливих тестів навіть для нескладних програмних компонент практично нескінченне, тому стратегія тестування полягає в тому, щоб провести всі можливі тести з урахуванням наявного часу та ресурсів. Як результат програмне забезпечення (ПЗ) тестується стандартним виконанням програми з метою виявлення багів (помилки або інших дефектів).

Тестування ПЗ може надавати об'єктивну, незалежну інформацію про якість ПЗ, ризики відмови, як для користувачів так і для замовників.

Тестування може проводитись, як тільки створено виконуваний код (навіть частково завершено). Процес розробки зазвичай передбачає коли та як буде відбуватися тестування. Наприклад, при поетапному процесі, більшість тестів відбувається після визначення системних вимог і тоді вони реалізуються в тестових програмах. На противагу цьому, відповідно до вимог гнучкої розробки ПЗ, програмування і тестування часто відбувається одночасно [15].

Під час проведення тестування програмного застосунку виявлено, що застосунок відображається однаково в найбільш популярних браузерах, таких як Internet Explorer, Opera, Mozilla Firefox, Chrome.

Перевірка програми на юзабіліті показала, що програмний застосунок є зручним для користувача, має зручну навігаційну панель та інтуїтивно зрозумілу логічну структуру.

Тестування програмного забезпечення відбувається за стандартним процесом. Завдання або кроки включають визначення тестового середовища, розробку тестових випадків, написання сценаріїв, аналіз результатів тестування та подання звітів про дефекти.

					ІК93.090БАК.005 ПЗ	Арк
Зм.	Лист	№ докум.	Підпис	Дата		49

Тестування може зайняти багато часу. Для невеликих збірок може бути достатньо ручного або спеціального тестування. Однак для великих систем інструменти часто використовуються для автоматизації завдань. Автоматизоване тестування допомагає командам реалізовувати різні сценарії, тестувати відмінності (наприклад, переміщення компонентів у хмарне середовище) і швидко отримувати відгуки про те, що працює, а що ні.

Хороший підхід до тестування охоплює інтерфейс прикладного програмування (API), інтерфейс користувача та рівні системи. Крім того, чим більше тестів буде автоматизовано та запущено раніше, тим краще. Деякі команди створюють власні засоби автоматизації тестування. Проте рішення постачальників пропонують функції, які можуть оптимізувати ключові завдання керування тестуванням, наприклад:

Постійне тестування: команди проекту тестують кожен збірник, щойно вона стає доступною. Цей тип тестування програмного забезпечення базується на автоматизованому тестуванні, інтегрованому в процес розгортання. Це дає змогу перевіряти програмне забезпечення в реалістичних тестових середовищах на ранніх етапах процесу, покращуючи дизайн і знижуючи ризики.

Керування конфігурацією: організації централізовано підтримують тестові активи та відстежують, яке програмне забезпечення збирається для тестування. Команди отримують доступ до таких активів, як код, вимоги, проектні документи, моделі, тестові сценарії та результати тестування. Хороші системи включають автентифікацію користувачів і журнали аудиту, щоб допомогти командам виконати вимоги відповідності з мінімальними адміністративними зусиллями.

Віртуалізація служби: середовища тестування можуть бути недоступними, особливо на ранніх стадіях розробки коду. Віртуалізація сервісів імітує служби та системи, які відсутні або ще не завершені, що дозволяє командам зменшити залежність і швидше тестувати. Вони можуть повторно використовувати, розгортати та змінювати конфігурацію для тестування різних сценаріїв без необхідності змінювати початкове середовище.

					ІК93.090БАК.005 ПЗ	Арк
Зм.	Лист	№ докум.	Підпис	Дата		50

Відстеження дефектів або помилок: моніторинг дефектів важливий як для команд тестування, так і для команд розробників для вимірювання та покращення якості. Автоматизовані інструменти дозволяють командам відстежувати дефекти, вимірювати їх масштаб і вплив, а також виявляти пов'язані з ними проблеми.

Показники та звітність: звіти та аналітика дозволяють членам команди ділитися статусом, цілями та результатами тестування. Розширені інструменти інтегрують показники проекту та представляють результати на інформаційній панелі. Команди швидко бачать загальний стан проекту та можуть контролювати зв'язки між тестуванням, розробкою та іншими елементами проекту.

Під час перевірки не виявлено помилок коду.

Висновки до 3 розділу

Основними учасниками взаємодії є Student (студент) і Teacher (викладач, викладач, лаборант тощо). Інші учасники: асистенти, куратори (студентські групи, мобільність студентів тощо), методисти, завідувачі кафедр/викладачі, заступники завідувачів профільних кафедр/викладачі (навчальна діяльність, педагогічна діяльність, виховна робота тощо).

Тестування програмного забезпечення – це процес технічного дослідження, призначений для виявлення інформації про якість продукту відносно контексту, в якому він має використовуватись. Техніка тестування також включає як процес пошуку помилок або інших дефектів, так і випробування програмних складових з метою оцінки.

Виконання тестових дій на початку циклу допомагає тримати тестування на передньому плані, а не залишатися позаду після розробки. Попередні тести програмного забезпечення також означають, що усунення дефектів обходиться дешевше.

Під час перевірки не виявлено помилок коду.

					ІК93.090БАК.005 ПЗ	Арк
Зм.	Лист	№ докум.	Підпис	Дата		51

4 ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ОТРИМАНИХ РЕЗУЛЬТАТІВ

4.1 Оцінка адекватності розробленої онлайн-платформи

Розглянемо основні екрани веб-додатку.

Головна сторінка (рис. 4.1).

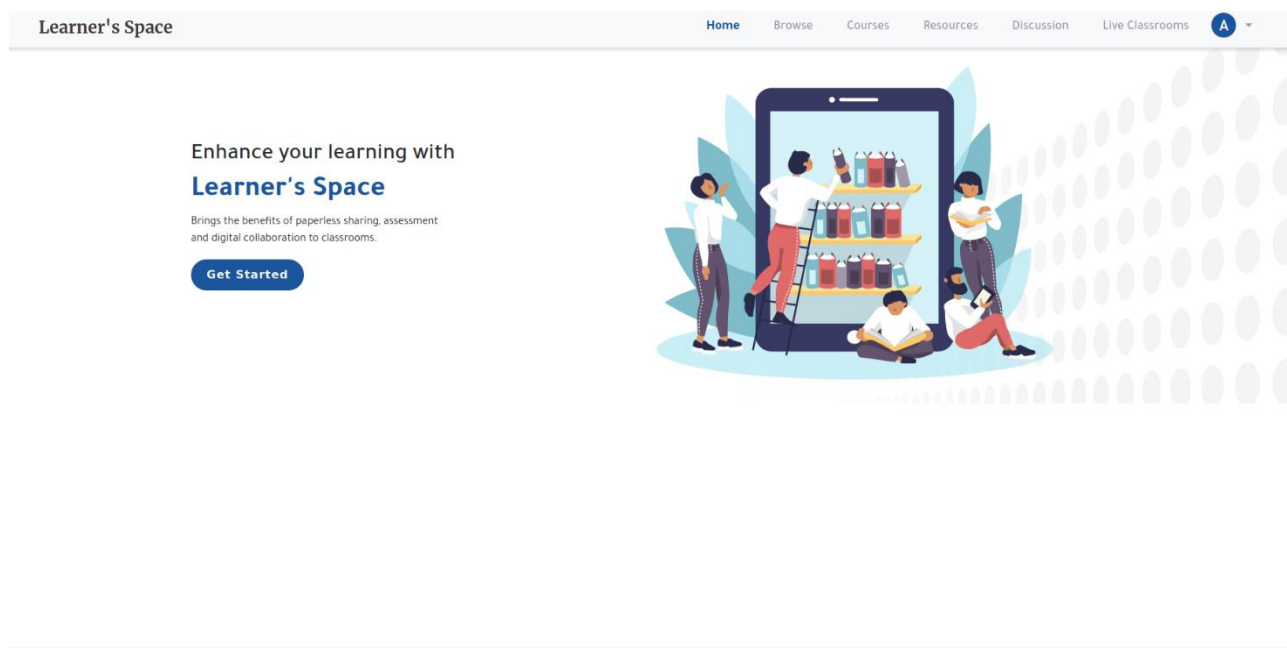


Рисунок 4.1 – Головна сторінка

Вікно реєстрації на онлайн-платформі (рис. 4.2).

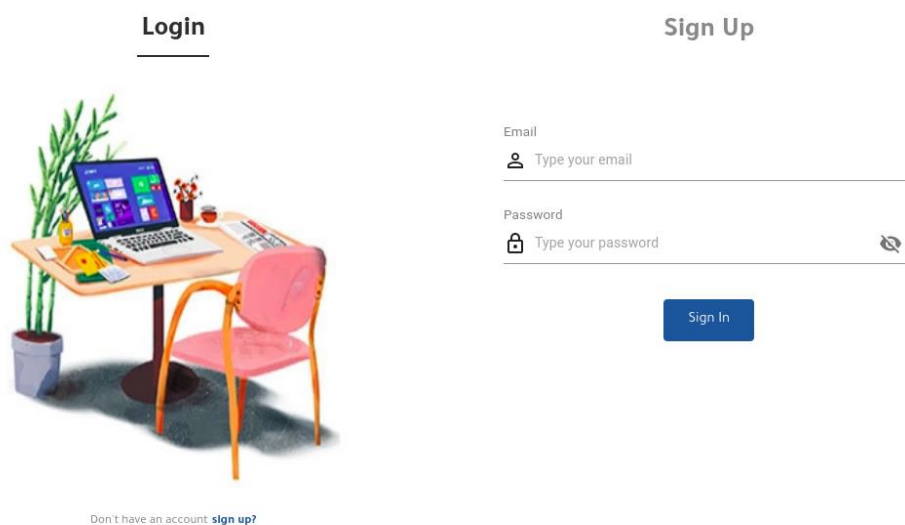


Рисунок 4.2 – Вікно реєстрації

					ІК93.090БАК.005 ПЗ	Арк
						52
Зм.	Лист	№ докум.	Підпис	Дата		

Введення особистих даних для реєстрації (рис. 4.3).

Login

Full Name

 Type your name

Email

 Type your email

Password

 Type your password 

Contact Number

 Type your Contact Number

Sign Up

Sign Up



Already registered [sign in?](#)

Рисунок 4.3 – Введення особистих даних

Відкривається особистий кабінет (рис. 4.4).

Learner's Space
Home **Browse** Courses Resources Discussion Live Classrooms A

Browse Learner's Space

Learn anytime, anywhere. Boost your skills flexible, online courses



Courses



Operating Systems
Computer Science
Basics of Operating Systems



Software Engineering
Computer Science
Basics of Software Engineering



Computer Networks
Computer Science
Basics of Computer Networks



MERN
Computer Science
Full Stack Development



MEAN
Computer Science
Full Stack Development



Soft Computing
Computer Science
Advanced soft computing

Рисунок 4.4 – Особистий кабінет

					ІК93.090БАК.005 ПЗ	Арк
Зм.	Лист	№ докум.	Підпис	Дата		53

Приєднання до нового класу навчання (рис. 4.5).

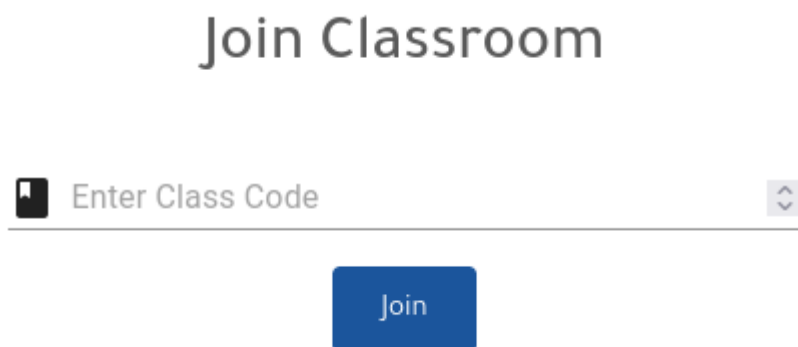


Рисунок 4.5 – Приєднання до класу

У класі відображається інформація про предмет, оцінки та завдання (рис. 4.6).

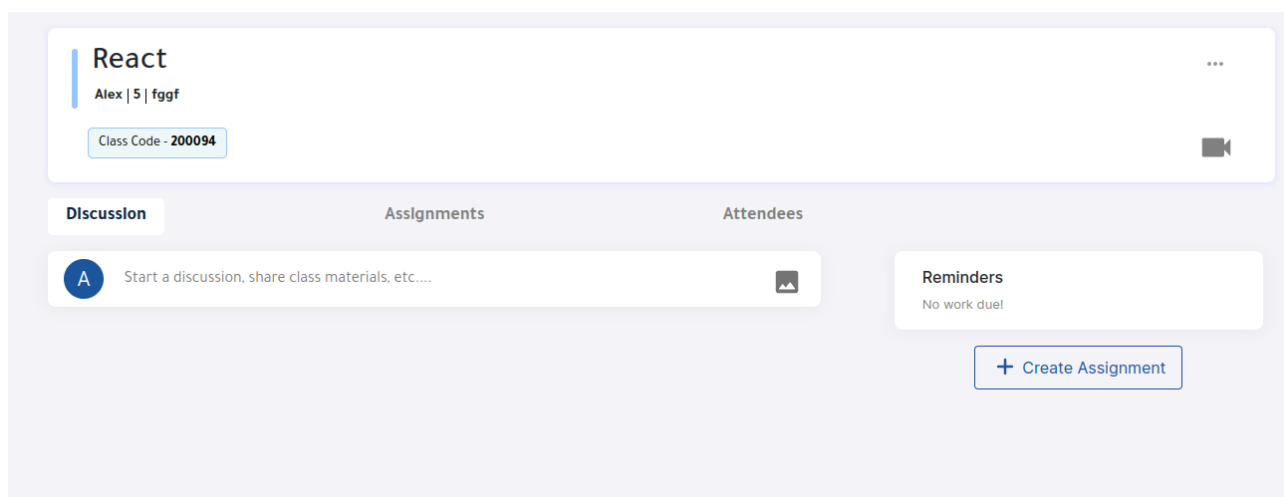


Рисунок 4.6 – Клас із завданнями

Також можна перейти в розділ «Обговорення», щоб задати питання викладачу або поспілкуватись із одногрупниками в класі.

					ІК93.090БАК.005 ПЗ	Арк
						54
Зм.	Лист	№ докум.	Підпис	Дата		

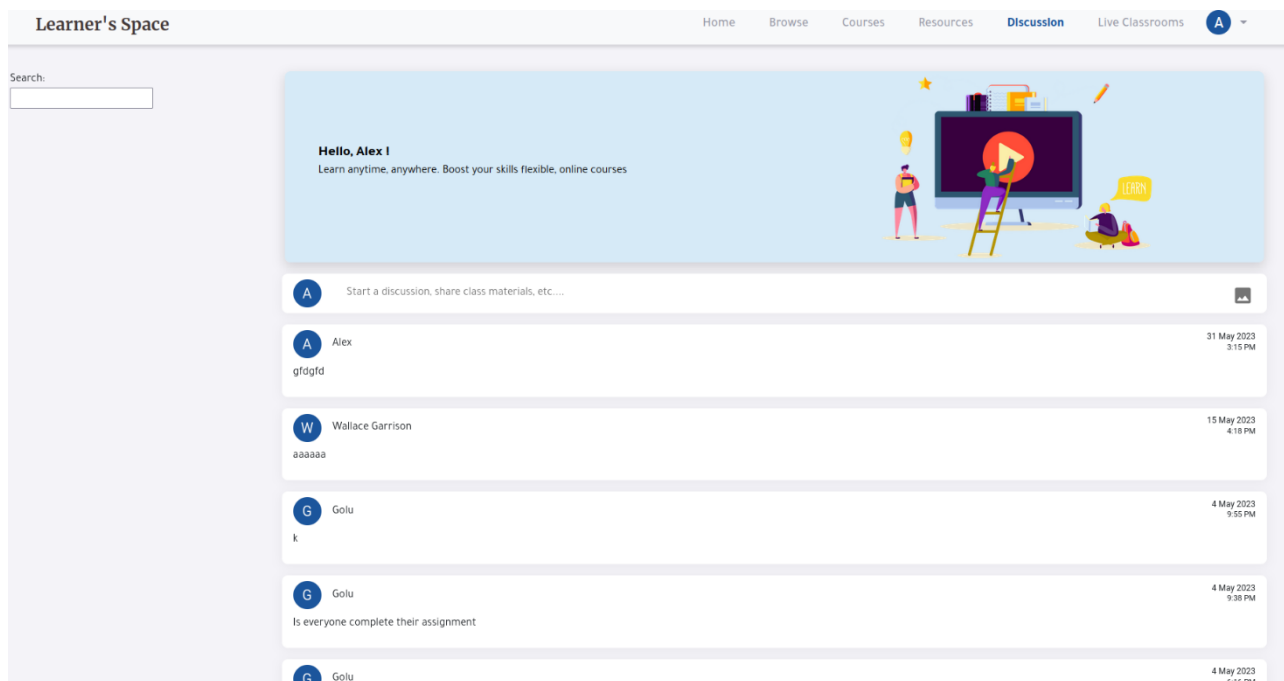


Рисунок 4.7 – Розділ «Обговорення»

4.2 Аналіз ефективності розробленого онлайн-платформи

Кандидати на іспит можуть працювати на комп'ютері звичним для них способом, а не ручкою та папером. Це означає, що процес навчання студентів знаходиться в невідгідному становищі, якщо вони не мають доступу до інструментів комп'ютерного редагування під час написання, наприклад, вирізати та вставляти текст або використовувати електронну таблицю для облікових записів. Важко перейти на ручку та папір, коли ви звикли працювати в цифровому вигляді з онлайн-курсом, і це може бути дуже втомливим протягом тривалого часу.

Організації значно зменшують адміністративний тягар щодо організації та проведення іспитів, коли вони переходять в Інтернет. Друк і розповсюдження екзаменів на папері, а також організація доставки готових сценаріїв до маркерів є трудомістким і дорогим процесом. Отже, онлайн-оцінювання та надання учням можливості скласти тест онлайн є більш своєчасними та ефективними.

					ІК93.090БАК.005 ПЗ	Арк
Зм.	Лист	№ докум.	Підпис	Дата		55

Набагато швидше оцінювати онлайн-оцінювання, і кандидати раді швидко отримати результати. За допомогою запитань, які автоматично оцінюються, студенти можуть отримати негайний зворотний зв'язок, а також постійний зворотний зв'язок із результатами своїх тестів.

Онлайн-навчання полегшує екзаменаційному органу керування всіма завданнями, пов'язаними зі створенням іспитів. Вперше за понад десять років авторські запитання можна створювати спільно, з чіткими робочими процесами для перегляду та затвердження запитань, перш ніж вони будуть додані до банку запитань. Це дозволяє викладачеві додавати нові знання та гарантувати, що вони дають студентам високоякісні тести здібностей, щоб допомогти кожному окремому учневі.

Після того, як буде схвалений банк запитань, екзаменаційні роботи можна легко створювати, вибираючи питання або використовуючи автоматизовані інструменти, напр. LOFT, або лінійне тестування на льоту, для створення автоматичних і рандомізованих паперів. Ви можете прочитати більше про лінійне тестування на льоту тут. Крім того, ці тести з варіантами відповідей можуть ефективно перевіряти знання учнів і покращувати їхній прогрес у навчанні так само, як і формувальне оцінювання. Поєднання тестових інструментів покращує індуктивне логічне мислення та покращує навчання студентів. Завдяки миттєвому відгуку про ці тести учні можуть справити найкраще враження завдяки швидшому вдосконаленню.

Завдяки онлайн-навчанню оцінювання та модерація результатів іспитів і тестів на здібності стало спрощеним і тепер систематичним процесом. Результати тестування пропонують швидкий зворотний зв'язок, оскільки вони набагато безпечніші та стандартизовані, коли ними керують онлайн.

Можливість проведення онлайн-тестування є набагато екологічнішою – загалом витрачається менше паперу, друку та транспорту.

					ІК93.090БАК.005 ПЗ	Арк
Зм.	Лист	№ докум.	Підпис	Дата		56

За допомогою онлайн-оцінювання екзаменаційні органи можуть пропонувати іспити кандидатам, які знаходяться в більшій географічній зоні, оскільки вони не обмежуються пропозицією іспитів у фізичних центрах тестування. Це дає змогу розвивати навчання студентів у всьому світі, що є початком масового навчання. Забезпечення того, щоб учні не обмежувалися лише тестами, які вони можуть пройти, підвищує шанси на освіту для тих, хто прагне подальшого розвитку освіти та нових знань.

Ще однією перевагою онлайн-оцінювання є більша безпека, оскільки всі онлайн-екзаменаційні роботи, відомості про кандидатів, оцінки та результати зберігаються в цифровому вигляді. Крім того, відповідний доступ до системи можуть отримати лише ті люди, які повинні мати доступ до інформації про онлайн-іспит. Набагато важче контролювати доступ, зберігання та результати керування за допомогою паперових іспитів.

Онлайн-оцінювання дозволяє швидко та чітко звітувати про результати та прогрес кандидатів. Це полегшує надання корисних відгуків кандидатам про їхні успіхи, сфери, у яких вони сильні, і які сфери навчання потребують уваги.

Онлайн-оцінювання є економічно ефективнішим завдяки значному скороченню адміністративного часу на керування всіма процесами створення іспиту, тестування та виставлення оцінок.

Висновки до 4 розділу

Конкурентоспроможність будь-якого вищого навчального закладу залежить від його здатності і уміння готувати кваліфікованих фахівців, якість яких не тільки задовольняє вимогам споживачів і всіх зацікавлених сторін, але і перевершує їх очікування.

					ІК93.090БАК.005 ПЗ	Арк
						57
Зм.	Лист	№ докум.	Підпис	Дата		

При перевірці знань, умінь і навичок велике значення має їх об'єктивна та незаангажована оцінка до якої висуваються наступні вимоги: оцінка повинна бути об'єктивною і справедливою, ясною і зрозумілою для студента, оцінка повинна виконувати стимулюючу функцію та орієнтувати на навчання, оцінка повинна бути всебічною та враховувати усі види навчальної діяльності.

					ІК93.090БАК.005 ПЗ	Арк
Зм.	Лист	№ докум.	Підпис	Дата		58

ВИСНОВКИ

Дослідження онлайн-платформи для аналізу успішності студентів є дуже актуальним у сучасній освітній системі. Такі платформи надають університетам, вищим навчальним закладам та навчальним програмам можливість збирати та аналізувати дані про студентів з метою вдосконалення процесу навчання та забезпечення кращої якості освіти.

Аналіз успішності студентів дозволяє ідентифікувати сильні та слабкі сторони навчальних програм і методик, щоб вчителі могли адаптувати свої підходи до потреб студентів. Це може сприяти покращенню результатів навчання та залученню студентів.

За допомогою аналізу даних, платформи можуть збирати інформацію про індивідуальні успіхи студентів, їхні пріоритети та потреби. Це дає можливість налаштувати навчальний процес на основі цих даних, забезпечуючи індивідуальну підтримку кожному студентові.

Збирання та аналіз даних про успішність студентів дозволяє виявляти загальні тренди та патерни, які можуть бути корисними для розуміння, як працюють навчальні програми та як їх можна вдосконалювати.

Онлайн-платформи дозволяють вимірювати ефективність різних навчальних стратегій та методик. Це дозволяє установам вищої освіти зробити обґрунтовані вибори щодо навчальних планів, ресурсів та методів навчання.

Дослідження успішності студентів на онлайн-платформах надає велику кількість даних, які можуть бути використані для вдосконалення навчальних програм, розробки нових курсів та стратегій, а також підтримки прийняття рішень на рівні установ вищої освіти.

Тому проблема, розглянута в цьому дипломному проекті, є актуальною та має широке практичне призначення як на сьогоднішній день, так і в майбутньому.

Метою даного проекту є розробка онлайн-платформи для аналізу успішності студентів.

					ІК93.090БАК.005 ПЗ	Арк
Зм.	Лист	№ докум.	Підпис	Дата		59

Системи дистанційного навчання призначені для навчання за межами навчальних місць. Мета використання систем це підвищення якості навчання, створення короткострокових курсів для підвищення кваліфікації учнів, а також для накопичення та застосування знань в різних наукових галузях науки, що досліджуються в різних навчальних закладах.

В своєму дипломному проекті я обираю мову JavaScript, давайте розглянемо його специфіку та переваги.

JavaScript є найбільш широко використовуваною мовою в області розробки в усьому світі. Вже більше 25 років JavaScript Frameworks керує Інтернетом. Але код JavaScript обережно оброблявся в фреймворках JavaScript принаймні протягом останнього десятиліття.

JavaScript є однією з найпопулярніших мов і в даний час є вибором як для фронтальної, так і для серверної веб-розробки. Крім того, у середині фреймворку JavaScript ви, ймовірно, виберете досить скоро.

Для забезпечення доступу до інформації із бази даних для кожної із таблиць потрібно створити entity клас на мові програмування Java з властивостям, які будуть відображати поля таблиці в базі даних.

Repository надає інтерфейс що б отримати доступ і маніпулювати даними, які є збереженими в базі даних.

Клас Entity є аналогом таблиці в базі даних, поле моделі являється відповідною копією значення окремого стовпця рядка.

Окрім цього при застосуванні об'єктно-реляційної проекції надається можливість зручного отримання даних із таблиць, які мають зв'язки «багато-до-багатьох» і «один-до-багатьох».

База даних системи реалізована за допомогою ORM Hiberante. Розглянемо більш детально структури основних таблиць бази даних.

					ІК93.090БАК.005 ПЗ	Арк
Зм.	Лист	№ докум.	Підпис	Дата		60

Технічні вимоги до комп'ютера, який відповідає мінімальним системним вимогам для розробки і розгортання створюваної програми, інсталяції інструментальних засобів розробки, що враховує особливості експлуатації програмного забезпечення, приведені для всіх використовуваних програмних засобів. У такому випадку для розробки сайту буде використовуватись Visual Studio Code.

Під час проведення тестування програмного застосунку виявлено, що застосунок відображається однаково в найбільш популярних браузерах, таких як Internet Explorer, Opera, Mozilla Firefox, Chrome.

Перевірка програми на юзабіліті показала, що програмний застосунок є зручним для користувача, має зручну навігаційну панель та інтуїтивно зрозумілу логічну структуру.

Тестування програмного забезпечення відбувається за стандартним процесом. Завдання або кроки включають визначення тестового середовища, розробку тестових випадків, написання сценаріїв, аналіз результатів тестування та подання звітів про дефекти.

Під час перевірки не виявлено помилок коду.

					ІК93.090БАК.005 ПЗ	Арк
Зм.	Лист	№ докум.	Підпис	Дата		61

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Classroom | Google for Education [Електронний ресурс]. – URL: classroom.google.com/
2. Ivelin Demirov. Learn JavaScript VISUALLY. 1st Edition, Publisher – Nai Inc. 2014
3. JQuery API Documentacion [Електронний ресурс]. – Режим доступу: <https://api.jqueryui.com/>
4. Kristina Chodorow. MongoDB: The Definitive Guide. 2013.
5. Kyle Banker, Peter Bakkum, Shaun Verch, Douglas Garrett, Tim Hawkins. MongoDB in Action. 2016
6. Mark Myers. A Smarter Way to Learn JavaScript: The New Tech-Assisted Approach That Requires Half the Effort. 1st Edition, Publisher – CreateSpace Independent Publishing Platform. 2014.
7. Moodle – Open-source learning platform | Moodle.org [Електронний ресурс]. – URL: <https://moodle.org/?lang=uk>.
8. Peter Membrey, Eelco Plugge, Tim Hawkins. The Definitive Guide to MongoDB: The NoSQL Database for Cloud and Desktop Computing. 2010.
9. What Is Google Apps For Education? | Ss. Cosmas & Damian Catholic School [Електронний ресурс]. – URL: <https://www.sscdschool.com/what-is-google-apps-for-education->
10. What is Use Case Diagram?- [Електронний ресурс]. – Режим доступу: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-usecase-diagram/>
11. Косик В. Використання мобільних пристроїв та планшетів на базі ОС Android в навчальному процесі / Вікторія Косик. [Електронний ресурс]. Режим доступу: <http://www.airo.com.ua/vikoristannya-mobilnih-pristroyiv-ta-planshetiv-na-bazi-os-android-v-navchalnomu-protsesi/>.

					ІК93.090БАК.005 ПЗ	Арк
						62
Зм.	Лист	№ докум.	Підпис	Дата		

12. Мобільне навчання стає дедалі більш популярним [Електронний ресурс]: (НОВИНИ) / Management.com.ua // НОВИНИ від 07.04.2011. – Режим доступу: <http://www.management.com.ua/news/?id=1329>

13. Принципи побудови та етапи проектування баз даних – [Електронний ресурс], — Режим доступу: http://stud.com.ua/35671/informatika/kontsepsiya_baz_danih

14. Система керування базами даних [електронний ресурс] // Режим доступу: http://uk.wikipedia.org/wiki/Система_керування_базами_даних – Назва з екрану.

15. Типи мобільних додатків. [Електронний ресурс] — Режим доступу: <https://smile-ukraine.com/ua/mobile-apps/mobile-apps-types>

16. Що таке JQuery? [Електронний ресурс]. – Режим доступу: <https://phpacademy.kiev.ua/uk/blog/what-is-jquery>

17. Щодо особливостей організації освітнього процесу під час карантину : лист М-ва освіти і науки України від 25 берез. 2020 р. № 1/9-176 // Інформ. зб. для освітян. – 2020. – № 4. – С. 71–72.

					ІК93.090БАК.005 ПЗ	Арк
Зм.	Лист	№ докум.	Підпис	Дата		63

ДОДАТОК 1

Онлайн-платформа для аналізу успішності студентів

Текст програмного коду

ІК93.090БАК.005 Д1

Аркушів 12

Київ 2023 р

App.js

```
import {BrowserRouter as Router, Route,Switch, Redirect} from 'react-router-dom';

import './App.css';
import React, {useEffect} from 'react';
import Home from './components/Home/Home';
import Browse from './components/Browse/browse';
import Courses from './components/Courses/Dashboard';
import Courseroom from './components/Courses/Classroom';
import CourseAssign from './components/Courses/Assignments'
import Content from './components/Courses/Contents'
import Resources from './components/Resources/Dashboard'
import Forum from './components/Forum/Dashboard'
import Dashboard from './components/Classroom/Dashboard/Dashboard';
import Classroom from './components/Classroom/Classroom';
import ScrollToTop from './components/partials/ScrollToTop/ScrollToTop';
import AssignmentAdmin from
'./components/Classroom/AssignmentAdmin/AssignmentAdmin';
import StudentSubmission from
'./components/Classroom/StudentSubmission/StudentSubmission';
import { useDispatch, useSelector } from 'react-redux';
import { AUTOLOGIN, selectUserData } from './reduxSlices/authSlice';
import Reminders from './components/partials/Header/MobileReminder'
import CircularProgress from '@material-ui/core/CircularProgress';
const App = () => {
  const userData = useSelector(selectUserData);
```

```

return (
  <div className=«app»>
    <ScrollToTop>
      <Router>
        {
          userData.loading ? (
            <div className=«col-12 d-flex justify-content-center align-items-
center» style={{height:»100vh}}>
              <CircularProgress size={80} className=«display-block»/>
            </div>
          ) : userData.token ? (
            <Switch>
              <Route path="/" component={Home} exact/>
              <Route path="/browse" component={Browse} exact/>
              <Route path="/courses" component={Courses} exact/>
              <Route path="/courses/:id" component={Courseroom} exact/>
              <Route path="/courses/:id/:tab" component={Courseroom} exact/>
              <Route path="/courses/:id/assignment/:assignId/admin"
component={CourseAssign} exact/>
              <Route path="/courses/:id/content/:contentId/admin"
component={Content} exact/>
              <Route path="/resources" component={Resources} exact/>
              <Route path="/forum" component={Forum} exact/>
              <Route path="/classes" component={Dashboard} exact/>
              <Route path="/classes/reminders" component={Reminders}/>
              <Route path="/classes/:id" component={Classroom} exact/>
              <Route path="/classes/:id/:tab" component={Classroom} exact/>
              <Route path="/classes/:id/assignment/:assignId/admin"
component={AssignmentAdmin} exact/>

```

```

        <Route path='/classes/:id/assignment/:assignId'
component={StudentSubmission} exact/>
        <Redirect to =«/» />
    </Switch>
) : (
    <Switch>
        <Route path="/" component={Home} exact/>
        <Route path='/browse' component={Browse} exact/>
        <Redirect to =«/» />
    </Switch>
)
}
</Router>
</ScrollToTop>
</div>
)
}

```

export default App

server.js

```

const express = require('express');
const mongoose = require('mongoose');
const cors = require('cors');
const path = require('path');

const app = express();

```

```
app.use(express.json());
```

```
app.use(cors());
```

```
const authRoutes = require('./routes/auth');
```

```
const classroomRoutes = require('./routes/classroom');
```

```
const courseRoutes = require('./routes/course');
```

```
const resourceRoutes = require('./routes/resource');
```

```
const forumRoutes = require('./routes/forum');
```

```
const browseRoutes = require('./routes/browse');
```

```
app.use('/auth', authRoutes);
```

```
app.use('/classes', classroomRoutes);
```

```
app.use('/courses', courseRoutes);
```

```
app.use('/resources', resourceRoutes);
```

```
app.use('/forum', forumRoutes);
```

```
app.use('/browse', browseRoutes);
```

```
// if ( process.env.NODE_ENV === «production» || 1) {
```

```
//   app.use(express.static(path.join(__dirname, «../client/build»)));
```

```
//   app.get(«*», (req, res) => {
```

```
//     res.sendFile(path.resolve(__dirname, '..', 'client', 'build', 'index.html'));
```

```
//   })
```

```
// }
```

```
app.use((req, res, next) => {
```

```
  const err = new Error(«Not Found»);
```

```
  err.statusCode = 404;
```

```
  next(err);
```

```
  })
```

```
  app.use((err, req, res, next) => {  
    console.log(err);  
    const status = err.statusCode || 500;  
    const message = err.message;  
    res.status(status).json( {message: message} );  
  })
```

```
  mongoose.connect('mongodb+srv://admin:987654321@cluster0.edmu2.mongodb.net/myFirstDatabase?retryWrites=true&w=majority',  
    {  
      useUnifiedTopology: true,  
      useNewUrlParser: true  
    })  
  .then(result => {  
    app.listen(process.env.PORT || 5000);  
    console.log(«Server started at port 5000»);  
    console.log(«http://localhost:5000»);  
  })  
  .catch(err => {  
    console.log(err);  
  })
```

```
// https://localhost:5000/auth/login -> post  
// https://localhost:5000/auth/signup -> put
```

ClassList.jsx

Никита Волов, [01.06.2023 11:04]

```
import React, { useState } from 'react';
import { useHistory } from 'react-router-dom';
import «./Dashboard.css»;
import Card1 from «../../assets/card1.svg»;
import Card2 from «../../assets/card2.svg»;
import Card3 from «../../assets/card3.svg»;
import Card4 from «../../assets/card4.svg»;
import Card5 from «../../assets/card5.svg»;
import { Link } from 'react-router-dom';

const ClassList = (props) => {
  const [show, setShow] = useState(false);
  const toggle = () => setShow(prevState=>!prevState);
  const browse = props.browse;

  // courses display part
  const RenderClasses = () => {

    return (
      (browse.map((sub,index) => {
        let backgroundStyle = {};
        let card = Card1;
        switch (index%5) {
          case 0 : {
            backgroundStyle =
{backgroundColor:»#E5F8F0»,borderRadius:»12px»};
            card=Card1;
```

```

    }
    break;
    case 1 : {
        backgroundStyle =
{backgroundColor:»#FFF3DA»,borderRadius:»12px»};
        card=Card2;
    }
    break;
    case 2 : {
        backgroundStyle =
{backgroundColor:»#E9E9FF»,borderRadius:»12px»};
        card=Card3;
    }
    break;
    case 3 : {
        backgroundStyle =
{backgroundColor:»#FFE9E9»,borderRadius:»12px»};
        card=Card4;
    }
    break;
    case 4 : {
        backgroundStyle =
{backgroundColor:»#FEE2FF»,borderRadius:»12px»};
        card=Card5;
    }
}
// console.log(history);
return (

```

```
<div key={index} className=«col-12 col-md-6 col-lg-4 d-flex justify-content-center»>
```

```
<Link to={«/signup»}  
style={{maxWidth:»100%»,minWidth:»100%}}}>
```

```
<div className=«d-none d-md-flex card class-card card-width mx-auto» style={backgroundStyle}>
```

```
<img src={card} className=«mx-auto mt-3» height=«130px»/>
```

```
<div className=«card-body m-3 mx-md-4 rounded-3»  
style={{backgroundColor:»#fff}}}>
```

```
<h5 className=«card-title heading-3 text-start mb-0»>{sub.Name}</h5>
```

```
<p className=«card-text text-dark mb-1»><span  
className=«text-muted» style={{fontSize:»16px}}>{sub.fieldName}</span></p>
```

```
<p className=«card-text text-dark mb-1»  
style={{fontSize:»14px}}>{sub.desc}</p>
```

```
</div>
```

```
</div>
```

```
</Link>
```

```
</div>
```

```
);
```

```
})
```

```
));
```

```
}
```

```
const RenderClassesMobile = () => {  
  return (
```

```

(browse.map((sub,index) => {
  let backgroundStyle ={};
  let card = Card1;
  switch (index%5) {
    case 0 : {
      backgroundStyle =
{backgroundColor:»#E5F8F0»,borderRadius:»12px»};
      card=Card1;
    }
    break;
    case 1 : {
      backgroundStyle =
{backgroundColor:»#FFF3DA»,borderRadius:»12px»};
      card=Card2;
    }
    break;
    case 2 : {
      backgroundStyle =
{backgroundColor:»#E9E9FF»,borderRadius:»12px»};
      card=Card3;
    }
    break;
    case 3 : {
      backgroundStyle =
{backgroundColor:»#FFE9E9»,borderRadius:»12px»};
      card=Card4;
    }
    break;
  }
})

```

```

        case 4 : {
            backgroundStyle =
{backgroundColor:»#FEE2FF»,borderRadius:»12px»};
            card=Card5;
        }
    }
    return (
        <div key={index} className=«col-12 col-md-6 col-lg-4 d-flex justify-
content-center card-width-mobile»>
            <Link to={«/browse/»} className=«card-width-mobile»>
                <div className=«d-block d-md-none card mb-4»
style={backgroundStyle}>
                    <div className=«row g-0»>
                        <div className=«col-4 d-flex justify-content-center»>
                            <img src={card} className=«img-fluid rounded-start px-2»
width=«100%»/>
                        </div>
                        <div className=«col-8 d-flex align-items-center»>
                            <div className=«card-body link-wrap»>
                                <h5 className=«card-title heading-3 text-start px-0 mb-
0»>{sub.Name}</h5>

                                <p className=«card-text text-dark mb-1»><span className=«text-
muted»>{sub.fieldName}</span></p>

                                <p className=«card-text text-dark mobile-card-desc mb-
1»>{sub.desc}</p>
                            </div>
                        </div>
                    </div>
                </div>
            </div>

```

```

        </div>
    </Link>
</div>

);
})
));
}

return (
    <
    <div className=«col-12»>
        <div className=«row my-3 ms-0 content-box d-none d-md-block pb-4»>
            <div className=«col-12 heading-2 d-flex py-3 pb-4 justify-content-
center»>
                Courses
            </div>
            <div className=«col-12»>
                <div className=«row g-md-4 gy-md-5 px-md-3»>
                    <RenderClasses/>
                </div>
            </div>
        </div>
        <div className=«row my-3 ms-0 d-block d-md-none»>
            <div className=«heading-2 ps-4»>Courses</div>
            <div className=«col-12 content-box py-4 mobile-classlist»>
                <div className=«row px-2 py-0 pt-2»>
                    <RenderClassesMobile/>
                </div>
            </div>
        </div>
    </

```

</div>

</div>

</>

);

}

export default ClassList;