

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

Факультет комп'ютерних наук та кібернетики
Кафедра теорії та технології програмування


Кваліфікаційна робота
на здобуття ступеня бакалавра
за спеціальністю 122 Комп'ютерні науки
на тему:

**ВЕБЗАСТОСУНОК ДЛЯ ПІДТРИМКИ НАВЧАЛЬНОГО
ПРОЦЕСУ**

Виконала студентка 4-го курсу
Олександра ОРДАК


(підпис)

Науковий керівник:
доцент, кандидат фіз.-мат. наук
Людмила ОМЕЛЬЧУК


(підпис)

Засвідчую, що в цій роботі немає запозичень з
праць інших авторів без відповідних посилань.

Студентка


(підпис)

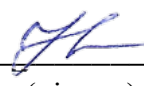
Роботу розглянуто і допущено до захисту на
засіданні кафедри теорії та технології
програмування

«05» червня 2023 р.,

протокол № 18

Завідувач кафедри

Микола НІКІТЧЕНКО


(підпис)

Київ – 2023

РЕФЕРАТ

Обсяг роботи 51 сторінка, основний текст роботи складає 47 сторінок, 36 ілюстрацій, 16 таблиць, 11 джерел посилань та 4 додатка.

ASP.NET CORE, BOOTSTRAP, ENTITY FRAMEWORK, БАЗА ДАНИХ, ВЕБЗАСТОСУНОК, ЗАКЛАД ВИЩОЇ ОСВІТИ.

Об'єктом роботи є процес взаємодії з навчальною платформою: перегляд, редагування, та збір інформації. Предметом роботи є вебзастосунок для налагодження та організації навчальних процесів в закладах вищої освіти (ЗВО).

Метою роботи є створення вебзастосунку для автоматизації процесу організації освітнього процесу в ЗВО з використанням технологій дистанційного навчання, що допомагає відстежувати успішність студентів і, крім того, служить джерелом необхідної інформації, як для студентів, так і для викладачів.

Методи розроблення: технології Entity Framework Core (EF Core) Database First – ORM Framework, ASP.NET Core MVC. Для організації бази даних використовувався SQL Server Management Studio (SSMS). Для стилізації сторінок було підключено Bootstrap. Логотип платформи розроблявся з допомогою інструменту, що має назву Canva. Середовище розробки проєкту – Microsoft Visual Studio. Мова програмування – C#.

Результати роботи: у ході розробки застосунку проведено аналіз доступних на ринку систем з подібним функціоналом, виконано роботу з базами даних та технологіями для розробки вебзастосунків, розроблено діаграми прецедентів, реалізовано застосунок, який дозволяє переглядати необхідну для навчального процесу інформацію. Програмний продукт «University Platform» призначений для використання адміністрацією ЗВО, викладачами, а також студентами цих ЗВО.

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧЕННЯ	5
ВСТУП	6
РОЗДІЛ 1. ОГЛЯД НАЯВНИХ НА РИНКУ СИСТЕМ.....	10
1.1. Платформа «Moodle».....	10
1.2. Платформа «Triton»	11
1.3. Система «Вищий навчальний заклад»	12
РОЗДІЛ 2. ОГЛЯД ВИКОРИСТАНИХ ТЕХНОЛОГІЙ, ПРИЗНАЧЕННЯ ТА ЦІЛІ СТВОРЕННЯ ЗАСТОСУНКУ	13
2.1. Огляд використаних технологій.....	13
2.2. Призначення застосунку	15
2.3. Цілі створення застосунку	15
2.4. Вимоги до вебзастосунку	16
2.4.1. Вимоги до структури та функціонування системи	16
2.4.2. Вимоги до функцій, які виконуються системою	17
2.4.3. Технічні вимоги	21
РОЗДІЛ 3. РЕАЛІЗАЦІЯ ВЕБЗАСТОСУНКУ	22
3.1. Опис організації бази даних.....	22
3.1.1. Логічна структура бази даних	22
3.1.2. Опис таблиць бази даних	23
3.2 Реалізація вебзастосунку.....	27
РОЗДІЛ 4. ІНСТРУКЦІЯ КОРИСТУВАЧА	32
4.1 Інструкція користувача в ролі студента	34
4.2 Інструкція користувача в ролі викладача	39
4.3 Інструкція користувача в ролі методиста.....	43
4.3 Інструкція користувача в ролі адміністратора.....	44
ВИСНОВКИ.....	46
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	47
ДОДАТКИ.....	48

Додаток А. Use case діаграма студента.....	48
Додаток Б. Use case діаграма викладача.....	49
Додаток В. Use case діаграма методиста	50
Додаток Г. Use case діаграма адміністратора.....	51

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧЕННЯ

API –	Application Programming Interface, прикладний програмний інтерфейс;
ASP.NET –	Active Server Pages Network Enabled Technologies, активні серверні сторінки мережових технологій;
EF –	Entity Framework, фреймворк сутностей;
HTML –	Hypertext Markup Language, мова розмітки гіпертексту;
MVC –	Model - View – Controller, представлення -вид- контролер;
ORM –	Object-relational mapping, об'єктно-реляційна проєкція;
SQL –	Structured query language, мова структурованих запитів;
SSMS –	SQL Server Management Studio, студія керування серверами SQL;
АСУ –	Автоматизована система керування;
БД –	База даних;
ЗВО –	Заклад вищої освіти;
ПІБ	Прізвище ім'я по батькові;
СУБД –	Система управління базами даних.

ВСТУП

Оцінка сучасного стану об'єкта розробки. Аналіз останніх досліджень і публікацій свідчить про стрімку зацікавленість закладами вищої освіти (ЗВО) багатьох країн світу в застосуванні інформаційних технологій для підтримки навчальних процесів. Автоматизація управління освітнім процесом, електронним документообігом, звітністю, та іншими рутинними процесами все більше обговорюється університетами і коледжами, які активно шукають шляхи для її впровадження [1].

Платформи, що використовуються у закладах вищої освіти (ЗВО) набули особливої затребуваності в період переходу на дистанційний формат викладання. Цей факт, та усі перелічені вище, дають підстави стверджувати, що цифрова трансформація освітніх процесів є незворотною.

Зазвичай такі платформи легкі в користуванні, мають доволі простий користувальницький інтерфейс та широкий спектр функціональних можливостей. Автоматизація багатьох аспектів робить його організацію швидшою та зрозумілішою для всіх стейкхолдерів освітнього процесу. Однак часто такі системи не містять усіх необхідних рішень для комфортного і доступного навчання, а інформація може бути розосереджена на різного виду платформах. До прикладу, більшість впроваджених аналогічних додатків не містять розкладу з посиланнями на пари, через що студентам доводиться кожного разу шукати його на інших доступних джерелах.

Актуальність теми. Тенденція до викладання у дистанційному форматі набрала популярності з початком пандемії у 2020 році. У той час це було скоріше необхідністю, ніж власною ініціативою навчальних закладів. Зважаючи на те, що формат з використанням технологій дистанційного навчання застосовується ЗВО і на сьогодні, створення платформи, яка б містила всю необхідну інформацію, централізовану в одному місці та

автоматизувала процес налагодження усіх організаційних моментів є досить актуальною задачею. Більше того, зовсім не виключення, що тенденція використання віддаленої форми навчання збережеться і тоді, коли зникне така необхідність.

Відповідно до статті 49 Закону України «Про вищу освіту» однією з основних форм здобуття вищої освіти є дистанційна форма здобуття освіти, яка визначається як: *«індивідуалізований процес здобуття освіти, що відбувається в основному за опосередкованої взаємодії віддалених один від одного учасників освітнього процесу в спеціалізованому середовищі, що функціонує на основі сучасних психолого-педагогічних та інформаційно-комунікаційних технологій»* [2].

В Київському національному університеті імені Тараса Шевченка наразі розробляються та готуються до затвердження «Положення про організацію освітнього процесу за дистанційною формою здобуття освіти у Київському національному університеті імені Тараса Шевченка» та «Положення про використання технологій дистанційного навчання у Київському національному університеті імені Тараса Шевченка». Таким чином, задача розробки вебзастосунку з підтримки використання дистанційного навчання у освітньому процесі ЗВО є актуальною задачею.

Тому було прийнято рішення розробити онлайн-платформу, яка б забезпечувала якісну організацію освітніх процесів та керування ними у ЗВО, де було б зібрано інформацію про студентів, викладачів, освітні компоненти, розклади, а також результати успішності студентів відповідно до форм контролю у єдиній системі.

Використовуючи такий вебзастосунок, можна суттєво покращити якість освітніх процесів та полегшити взаємодію користувачів з навчальним закладом.

Мета й завдання роботи. Метою даної роботи є розробка вебзастосунку для автоматизації процесу організації освітнього процесу з використанням технологій дистанційного навчання в ЗВО, що дозволяє відстежувати успішність студентів. Щоб досягти поставленої мети були визначені такі завдання:

- провести аналіз предметної області;
- розробити діаграми прецедентів для усіх видів ролей та діаграму бази даних;
- створити базу даних в SQL Server Management Studio (SSMS);
- стилізувати візуальну частину проєкту;
- налаштувати фільтри пошуку;
- реалізувати формування та збереження звітів у форматі Excel (експорт та імпорт файлів);
- реалізувати автентифікацію, авторизацію, керування користувачами та їх ролями.

Об'єкт, методи й засоби розроблення. Об'єктом роботи є організація освітнього процесу у ЗВО з використанням технологій дистанційного навчання. Предметом роботи є вебзастосунок для налагодження та організації освітнього процесу в ЗВО з використанням технологій дистанційного навчання.

Під час розробки вебзастосунку використовувались такі технології, як: Entity Framework Core (EF Core) Database First – ORM Framework та ASP.NET MVC. Для організації бази даних використовувався SQL Server Management Studio (SSMS). Для стилізації сторінок було підключено Bootstrap. Логотип платформи розроблявся з допомогою інструменту, що має назву Canva. Середовищем розробки проєкту є Visual Studio від Microsoft. Мова програмування, на якій розроблялась платформа – C#.

Можливі сфери застосування. Програмний продукт «University Platform» розроблений для використання у закладах вищої освіти з метою вирішення організаційних питань під час проведення навчання в онлайн-форматі. Застосунок може полегшити взаємодію різних категорій стейкхолдерів освітнього процесу із ЗВО незалежно від формату навчання, тому його доцільно використовувати і для очної форми здобуття вищої освіти.

РОЗДІЛ 1. ОГЛЯД НАЯВНИХ НА РИНКУ СИСТЕМ

1.1. Платформа «Moodle»

Однією з конкуруючих систем на ринку є платформа для управління навчальним процесом «Moodle» [3]. Вигляд сайту навчальної платформи Moodle наведено на рис. 1.

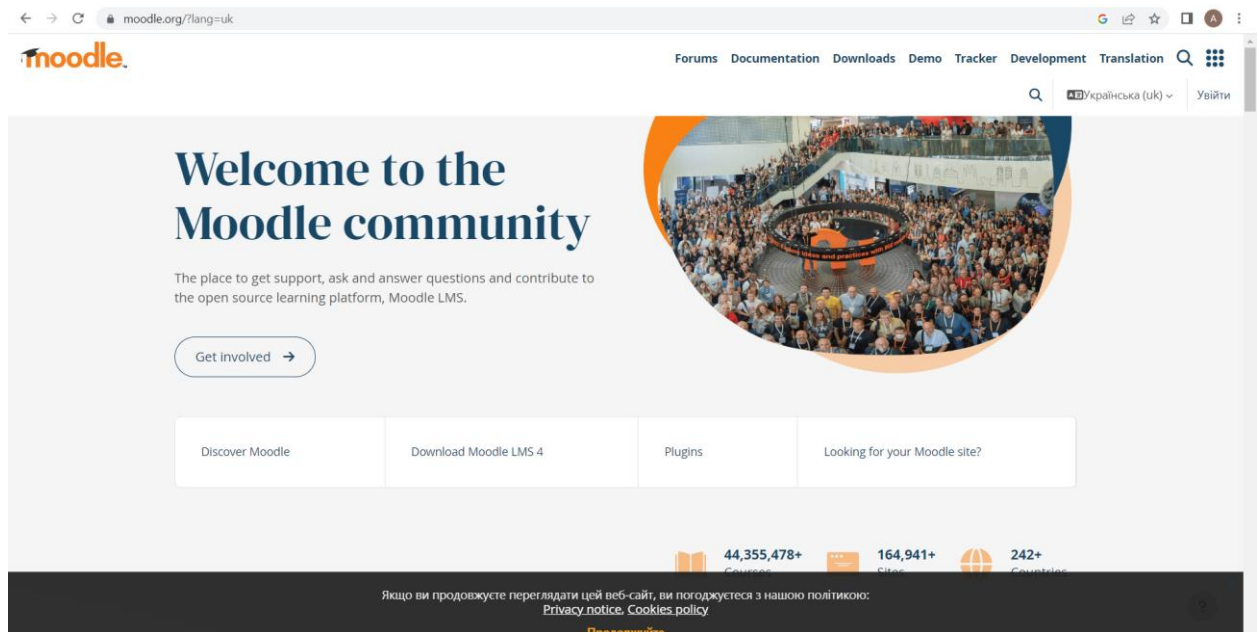


Рисунок 1 – Сайт навчальної платформи Moodle

Даний продукт є досить популярним і відомим на ринку [3]. До переваг слід віднести широку функціональність та наявність мобільної версії системи. Також «Moodle» дозволяє обмінюватись повідомленнями між студентами та викладачами та надає можливість завантажувати на платформу виконані роботи, прикріпляти файли, створювати курси та тестування.

Основним недоліком платформи «Moodle» є відсутність розкладу з посиланнями на пари.

1.2. Платформа «Triton»

Інша платформа, призначена для підтримки освітнього процесу в Київському національному університеті імені Тараса Шевченка – платформа «Triton» [4].

На рис. 2 зображено головну сторінку цього ресурсу.



Рисунок 2 – Сайт навчальної платформи «Triton Student»

Серед переваг платформи можна виокремити наступні:

- реєстрація та авторизація кожного студента індивідуально;
- наявність графіка контролю дисциплін та навчального плану освітньої програми студента.

До недоліків платформи можна віднести:

- відсутність навчального розкладу для кожної з груп;
- відображення лише остаточних оцінок за курс, що не дає в повній мірі прослідкувати успішність студентів з певного предмету.

1.3. Система «Вищий навчальний заклад»

Ще одним прикладом навчальної платформи є АСУ «Вищий навчальний заклад», що вже більше 10 років на ринку (див. рис. 3) [5, 6].

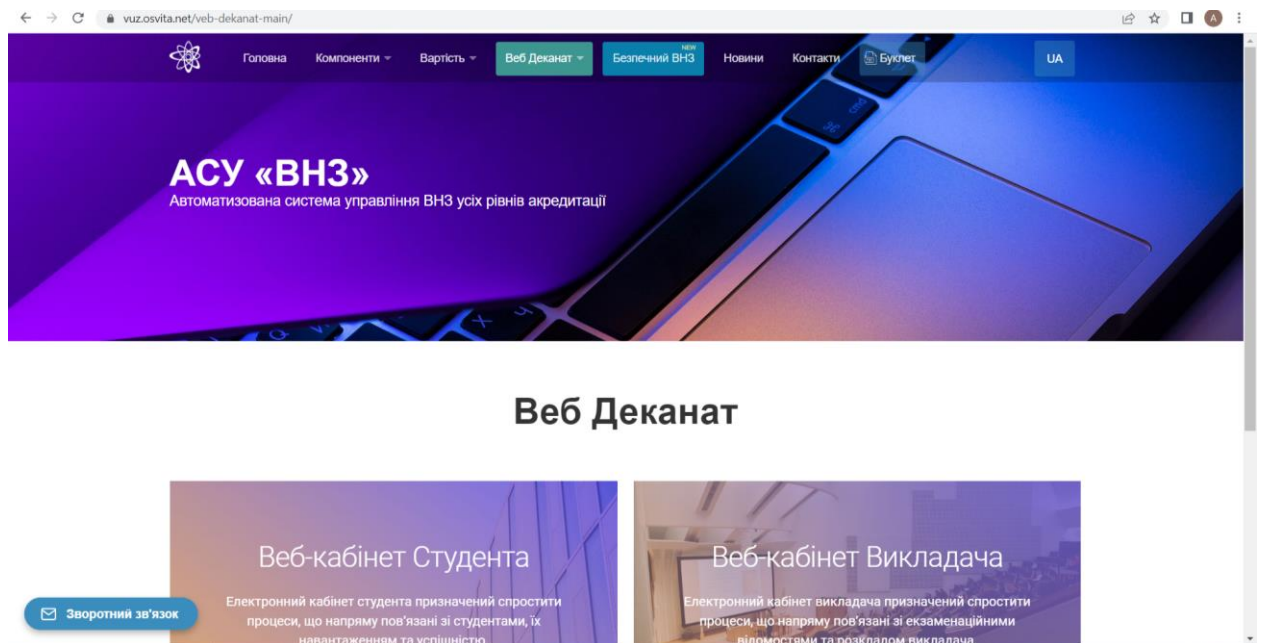


Рисунок 3 – Головна сторінка сайту АСУ «ВНЗ»

Це програмне забезпечення має розвинену систему захисту, користується популярністю серед багатьох університетів України. Серед переваг також можна виділити автоматизацію процесу роботи приймальної комісії.

З недоліків системи можна виділити те, що вона більше орієнтована на адміністрацію ЗВО та деканат, ніж на студентів.

Що стосується розробленої системи «University Platform», варто відзначити, що основними перевагами є зручність та простота використання, наявність розкладу для кожної групи з зазначеними посиланнями на пари, автоматизований процес зміни та додавання розкладу, здатність використовувати фільтри пошуку, експорт та імпорт файлів задля маніпулювання даними, а також можливість контролювати успішність студента за певними видами контролю з кожного предмету.

РОЗДІЛ 2. ОГЛЯД ВИКОРИСТАНИХ ТЕХНОЛОГІЙ, ПРИЗНАЧЕННЯ ТА ЦІЛІ СТВОРЕННЯ ЗАСТОСУНКУ

2.1. Огляд використаних технологій

При розробці даного проекту використовувався нижчеперелічений набір технологій:

- Entity Framework Core (EF Core) Database First – ORM Framework [7];
- ASP.NET MVC [8, 9];
- Bootstrap [10].

Entity Framework Core (EF Core) – це доступна для різних платформ версія технології доступу до даних Entity Framework. Дана технологія є ORM-інструментом. EF Core дозволяє працювати з даними незалежно від бази даних. Таким чином, на фізичному рівні ми працюємо з таблицями, первинними і зовнішніми ключами, а на концептуальному – вже з об'єктами.

Однією з переваг Entity Framework Core є підтримка різноманітних систем баз даних. Відповідно, завдяки EF Core можна працювати з будь-якої СУБД, у разі відсутності необхідності для неї провайдера.

Як технологія доступу до даних, Entity Framework Core має здатність до використання на різних платформах .NET. Це і стандартні платформи, такі як Windows Forms, і консольні застосунки: WPF, ASP.NET Core. При цьому багатоплатформність EF Core дозволяє взаємодіяти з нею не тільки на ОС Windows, але і на Linux і Mac OS X (на противагу EF 6).

Платформа ASP.NET MVC є фреймворком для створення сайтів і вебзастосунків, завдяки реалізації паттерна MVC [11].

Основною ідеєю паттерна MVC (Model-View-Controller), зображеного на рис. 4, є розбиття застосунка на три компоненти:

Контролер (controller) являє собою клас, що забезпечує зв'язок між користувачем і системою, представленням і сховищем даних. Він отримує введені користувачем дані і обробляє їх. І в залежності від результатів обробки відправляє користувачеві певний вивід, наприклад, у вигляді представлення.

Представлення (view) – представляє візуальну частину, власне, сам інтерфейс користувача програми.

Модель (model) представляє клас, що описує логіку даних, що використовуються.

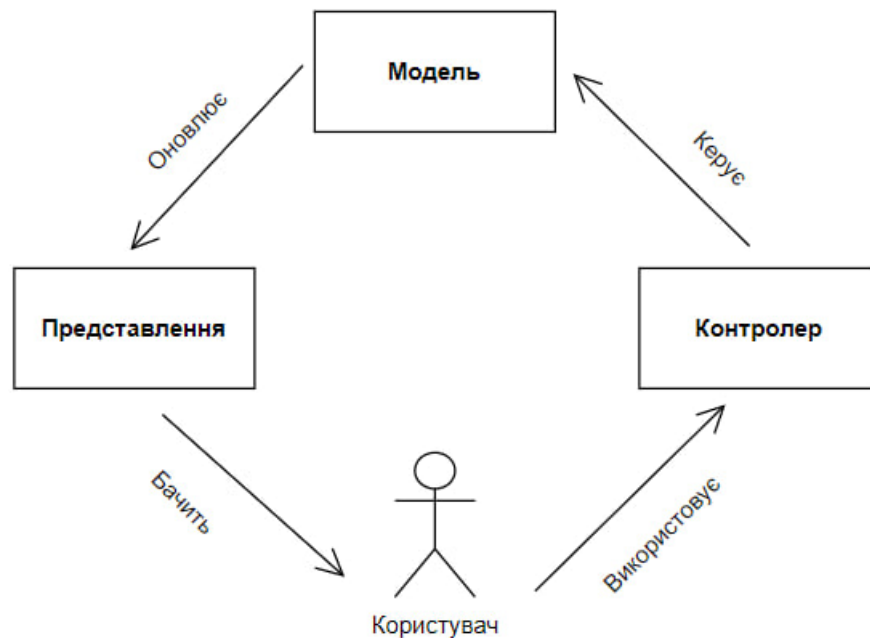


Рисунок 4 – Діаграма взаємодії шаблону проектування MVC

До основних плюсів фреймворка відносяться можливість швидко розробляти складні проєкти та відсутність обмежень у виборі СУБД.

Bootstrap – один з найпопулярніших CSS фреймворків від компанії Twitter для адаптивної верстки сайтів. Bootstrap – це набір готових CSS файлів, який можна підключити до сайту, задля швидкого і зручного застосування стилів та для правильного відображення сайту на різних видах девайсів.

Основа Bootstrap — модульна сітка, що складається з 12 колонок. Ця сітка має здатність перелаштовуватися відповідно до розміру екрану пристрою, з якого зайшов користувач на сайт.

Більше того, Bootstrap включає в себе стилі оформлення чималої кількості елементів сучасних сайтів, таких як: кнопки, форми, спливаючі вікна, навігаційні панелі і тому подібне.

Значуща перевага Bootstrap закладена в його поширеності. Його використовуванисть протягом довгого періоду часу і на великій кількості сайтів гарантує, що більшість проблем, пов'язаних з сумісністю браузерів та операційними системами, у ньому вирішені.

2.2. Призначення застосунку

Призначенням даного вебзастосунку є автоматизація процесу організації навчальних робіт в університетах в режимі онлайн та надання користувачам змоги взаємодіяти з платформою в найкомфортніших для нього умовах: отримувати доступ до потрібної інформації в будь-який час і в будь-якому місці (див. додаток А, додаток Б, додаток В, додаток Г).

Кваліфікаційна робота передбачає:

- аналіз методів, методик і моделей, що застосовуються для розв'язання задач автоматизації збору даних;
- аналіз наявних програмних засобів в галузі створення вебзастосунку;
- проєктування та програмну реалізацію системи «University Platform»;
- апробацію вебзастосунку «University Platform».

2.3. Цілі створення застосунку

Вебзастосунок «University Platform» розроблено з метою:

- забезпечення збору, обробки та аналізу інформації про студентів, викладачів та предмети (див. додаток А);

- відслідковування успішності студентів упродовж усього навчання в університеті;
- забезпечення можливості швидко і доступно переглянути розклад і зайти на пару через посилання;
- для викладачів: оприлюднення результатів студентів за роботи з різних видів контролю.

Також система призначена для редагування, збору, та аналізу інформації про користувачів (себто, студентів та викладачів), а також внесення змін у списки та розклад методистами (див. додаток В).

2.4. Вимоги до вебзастосунку

2.4.1. Вимоги до структури та функціонування системи

Застосунок повинен мати функціонал, який забезпечує реєстрацію та вхід для користувачів і адміністраторів.

В програмі передбачається виділити наступні ролі користувачів:

- роль методиста, що має змогу вносити зміни, видаляти та додавати інформацію в розділах, імпортувати та експортувати файли, вносити зміни у розклад (див. табл. 3);
- роль студента, який може лише переглядати інформацію в списках розділів, а також переходити за посиланням на заняття (див. табл. 1).
- роль викладача, який аналогічно студенту, має змогу переглядати інформацію у розділах платформи, переходити за посиланнями, завантажувати файли, але разом з тим має здатність керувати розділом «Бали»: додавати, редагувати та видаляти за необхідності бали студентів; а також має доступ до розділу «Форми контролю», у якому може самостійно вносити форми контролю, що використовуються для контролю успішності студентів з певного предмету (див. табл. 2).

- роль адміністратора, завдяки якій здійснюється керування користувачами та їх ролями (див. табл. 4);

2.4.2. Вимоги до функцій, які виконуються системою

Функції, що доступні для взаємодії з платформою для студента, а також їх задачі наведено у табл. 1.

Таблиця 1 – Перелік функцій та задач, що підлягають автоматизації для ролі студента

Функція	Задача
Реєстрація	Введення інформації про себе, електронної пошти та пароля
Вхід	Введення електронної пошти та пароля
Робота зі списком груп	Перегляд усіх груп
	Перегляд розкладу конкретної групи
	Перегляд студентів конкретної групи
Робота зі списком студентів	Перегляд усіх студентів
	Фільтрація за ПІБ студента
	Перегляд балів конкретного студента
Робота з списком балів конкретного студента	Перегляд
	Фільтрація за предметом та формою контролю
Робота зі списком викладачів	Перегляд усіх викладачів
	Перегляд предметів, що викладає конкретний викладач
Робота зі списком навчальних предметів	Перегляд усіх предметів
	Перегляд викладачів конкретного предмету
	Перегляд форм контролю конкретного предмету
Робота з розкладом занять	Перегляд

Дії викладача на платформі зазначено у табл. 2.

Таблиця 2 – Перелік функцій та задач, що підлягають автоматизації для ролі викладача

Функція	Задача
Реєстрація	Введення інформації про себе, електронної пошти та пароля
Вхід	Введення електронної пошти та пароля
Робота зі списком курсів	Перегляд списку груп конкретного курсу
Робота зі списком груп	Перегляд усіх груп
	Перегляд розкладу конкретної групи
	Перегляд студентів конкретної групи
	Завантаження списків груп з платформи
Робота зі списком студентів	Перегляд усіх студентів
	Фільтрація за ПІБ студента
	Перегляд балів конкретного студента
Робота зі списком балів	Перегляд
	Фільтрація за предметом, формою контролю та ПІБ студента
	Додавання нового балу
	Редагування балу
	Видалення балу
Робота зі списком викладачів	Перегляд усіх викладачів
	Перегляд предметів, що викладає конкретний викладач
Робота зі списком навчальних предметів	Перегляд усіх предметів
	Перегляд викладачів конкретного предмету
	Перегляд форм контролю конкретного предмету
Робота з розкладом пар	Перегляд

Робота з розкладами	Перегляд розкладу конкретної групи
Робота з формами контролю	Перегляд
	Створення нової форми контролю
	Редагування існуючої форми контролю
	Видалення форми контролю

У табл. 3. перелічено функціонал і задачі, доступні для ролі методиста.

Таблиця 3 – Перелік функцій та задач, що підлягають автоматизації для ролі методиста

Функція	Задача
Реєстрація	Введення інформації про себе, електронної пошти та пароля
Вхід	Введення електронної пошти та пароля
Робота зі списком курсів	Переглянути список груп конкретного курсу
	Редагування списку курсів
Робота зі списком груп	Перегляд усіх груп
	Перегляд розкладу конкретної групи
	Перегляд студентів конкретної групи
	Завантаження списків груп з платформи
	Завантаження списку групи на платформу
	Редагування списків груп
	Додавання нової групи
	Видалення групи
Робота зі списком студентів	Перегляд усіх студентів
	Фільтрація за ПІБ студента
	Перегляд балів конкретного студента
	Редагування
	Додавання нових студентів

	Видалення студентів
Робота зі списком балів	Перегляд
	Фільтрація за предметом, формою контролю та ПБ студента
Робота зі списком викладачів	Перегляд усіх викладачів
	Перегляд предметів, що викладає конкретний викладач
	Редагування списку викладачів
	Додавання нового викладача
	Видалення викладача
Робота зі списком навчальних предметів	Перегляд усіх предметів
	Перегляд викладачів конкретного предмету
	Перегляд форм контролю конкретного предмету
	Редагування предмету
	Додавання нового предмету
	Видалення предмету
Робота з розкладами	Перегляд розкладів
	Перегляд розкладу конкретної групи
	Редагування розкладу
	Додавання нового розкладу
	Видалення розкладу
Робота з розкладом пар	Перегляд
	Редагування
Робота з формами контролю	Перегляд

У табл. 4 зазначено функціонал адміністратора платформи.

Таблиця 4 – Перелік функцій та задач, що підлягають автоматизації для ролі адміністратора

Функція	Задача
Вхід	Введення електронної пошти та пароля
Управління роботою користувачів системи	Керування ролями
	Перегляд списку користувачів
	Надання прав доступу та визначення ролей для користувачів

2.4.3. Технічні вимоги

Браузери: сайт повинен коректно відображатися в інтернет-браузерах MS Internet Explorer 5.5 і вище, Mozilla 1.7 і вище, Opera 7.54 і вище.

На довільні некоректні дії користувача, як-от введення невірних даних, незаповнення обов'язкових полів та інші, які можуть бути оброблені системою, генеруються відповідні повідомлення про помилки українською мовою, в межах загального дизайну вебзастосунку.

РОЗДІЛ 3. РЕАЛІЗАЦІЯ ВЕБЗАСТОСУНКУ

3.1. Опис організації бази даних

3.1.1. Логічна структура бази даних

База даних проєкту була розроблена в SQL Server Management Studio (SSMS) (див. рис. 5).

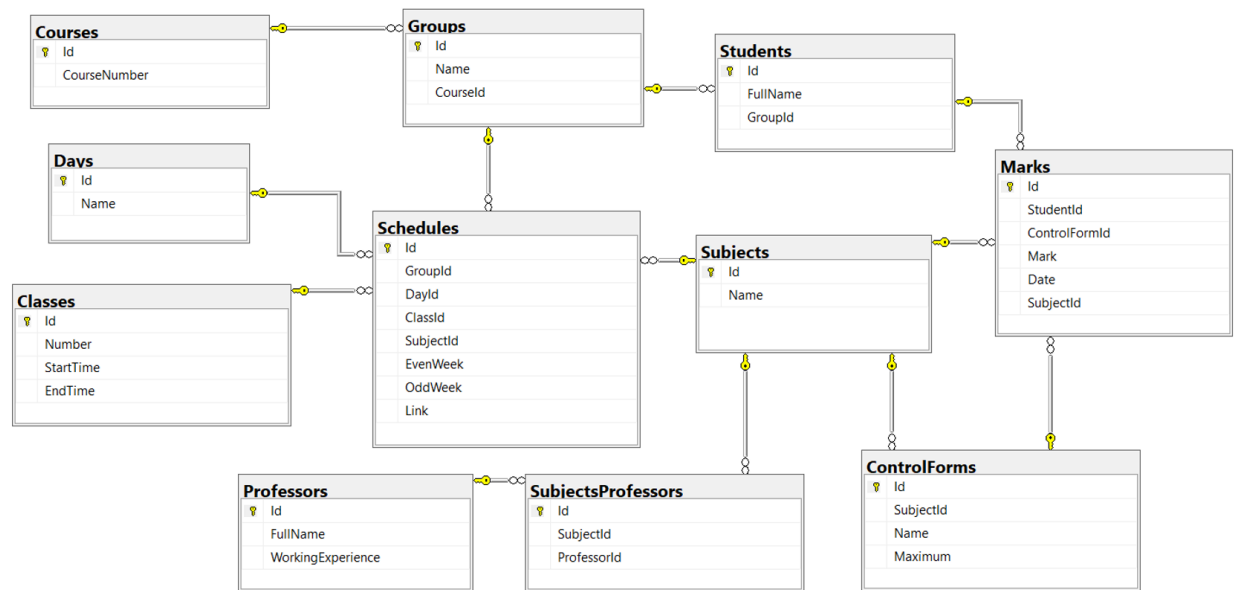


Рисунок 5 – Діаграма бази даних «DBUniversityPlatform»

В табл. 5 наведено перелік таблиць бази даних «DBUniversityPlatform». Всього таблиць у БД - 11.

Таблиця 5 – Перелік таблиць

Номер	Таблиця	Опис
1	Classes	Таблиця для збереження інформації про пари
2	ControlForms	Таблиця для збереження інформації про форми контролю
3	Courses	Таблиця для збереження інформації про курси
4	Days	Таблиця для збереження інформації про дні

5	Groups	Таблиця для збереження інформації про групи
6	Marks	Таблиця для збереження інформації про бали
7	Professors	Таблиця для збереження інформації про викладачів
8	Schedules	Таблиця для збереження інформації про розклади
9	Students	Таблиця для збереження інформації про студентів
10	Subjects	Таблиця для збереження інформації про предмети
11	SubjectsProfessors	Таблиця для збереження відповідності між предметом та викладачем

3.1.2. Опис таблиць бази даних

Структура кожної з таблиць БД описана у табл. 6 –16, де зазначено атрибут таблиці, його тип та опис.

Табл. 6 відповідає таблиці БД «Classes», що містить інформацію про пари: номер, час початку та час закінчення.

Таблиця 6 – Опис таблиці Classes

Атрибут	Тип	Опис
Id	int	Ідентифікатор
Number	int	Номер пари
StartTime	nvarchar(50)	Час початку пари
EndTime	nvarchar(50)	Час закінчення пари

Таблиця «ControlForms» (див. табл. 7) містить інформацію стосовно форм контролю окремих дисциплін, а саме: предмет, назва форми контролю, максимальна можлива кількість балів.

Таблиця 7 – Опис таблиці ControlForms

Атрибут	Тип	Опис
Id	int	Ідентифікатор
SubjectId	int	Предмет
Name	nvarchar(MAX)	Назва форми контролю
Maximum	int	Максимальна кількість балів

Таблиця «Courses» (див. табл. 8) містить номери курсів.

Таблиця 8 – Опис таблиці Courses

Атрибут	Тип	Опис
Id	int	Ідентифікатор
CourseNumber	int	Номер курсу

Таблиця «Days» (див. табл. 9) містить назви днів тижня.

Таблиця 9 – Опис таблиці Days

Атрибут	Тип	Опис
Id	int	Ідентифікатор
Name	nvarchar(50)	Назва дня

У таблиці «Groups» (див. табл. 10) збережені дані про групи (їх назва та курс).

Таблиця 10 – Groups

Атрибут	Тип	Опис
Id	int	Ідентифікатор
Name	nvarchar(MAX)	Назва групи
CourseId	int	Курс

Таблиця «Marks» (див. табл. 11) зберігає інформацію про бали: форму контролю за яку виставлено бал, предмет, дату, ПІБ студента, та сам бал.

Таблиця 11 – Опис таблиці Marks

Атрибут	Тип	Опис
Id	int	Ідентифікатор
StudentId	int	Студент
ControlFormId	int	Форма контролю
Mark	int	Бал
Date	nvarchar(50)	Дата
SubjectId	int	Предмет

Таблиця «Professors» (див. табл. 12) зберігає дані про викладачів, а точніше їх ПІБ та трудовий стаж.

Таблиця 12 – Опис таблиці Professors

Атрибут	Тип	Опис
Id	int	Ідентифікатор
FullName	nvarchar(MAX)	Прізвище, ім'я, по батькові викладача
WorkingExperience	int	Трудовий стаж

У таблиці «Schedules» (див. табл. 13) містяться записи стосовно навчальних розкладів, де зазначено групу, день, номер пари, предмет, тиждень (парний чи непарний), а також посилання на пару.

Таблиця 13 – Опис таблиці Schedules

Атрибут	Тип	Опис
Id	int	Ідентифікатор
GroupId	int	Група
DayId	int	День
ClassId	int	Пара
SubjectId	int	Предмет
EvenWeek	bit	Парний тиждень
OddWeek	bit	Непарний тиждень
Link	nvarchar(MAX)	Посилання на пару

Таблиця «Students» (див. табл. 14) зберігає інформацію про студентів: їх ПІБ та групу.

Таблиця 14 – Опис таблиці Students

Атрибут	Тип	Опис
Id	int	Ідентифікатор
FullName	nvarchar(MAX)	Прізвище, ім'я, по батькові студента
GroupId	int	Група

Таблиця «Subjects» (див. табл. 15) містить назви предметів, що викладаються.

Таблиця 15 – Опис таблиці Subjects

Атрибут	Тип	Опис
Id	int	Ідентифікатор
Name	nvarchar(MAX)	Назва предмета

У таблиці «SubjectsProfessors» наявна інформація про предмет та викладачів, що його викладають (див. табл. 16).

Таблиця 16 – Опис таблиці SubjectsProfessors

Атрибут	Тип	Опис
Id	int	Ідентифікатор
SubjectId	int	Предмет
ProfessorId	int	Викладач

3.2 Реалізація вебзастосунку

Первинною задачею в ході розробки вебзастосунку (після створення бази даних) було створення нового проєкту у середовищі Visual Studio 2022, і під'єднання попередньо розробленої БД.

Для цього були підключені необхідні пакети та виконано команду у Package Manager Console, зміно файл appsettings.json та метод ConfigureServices в файлі Startup.cs.

Після підключені до бази даних створились нові класи, які було переміщено у папку Models (див. рис. 6).

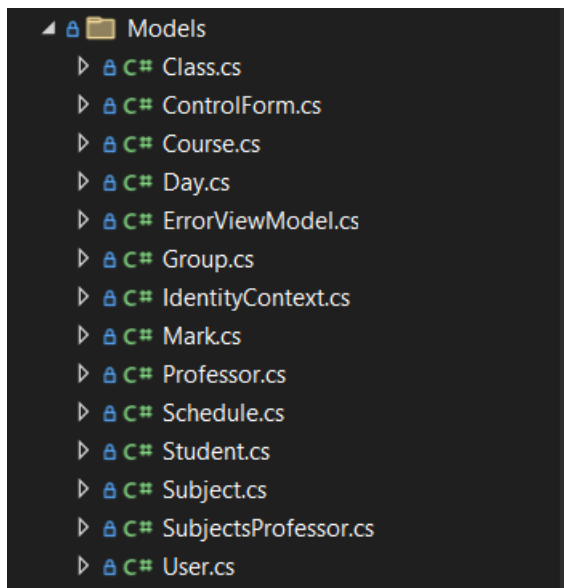


Рисунок 6 – Папка Models

Наступним кроком було створення та налагодження контролерів (див. рис. 7) та їх представлень.

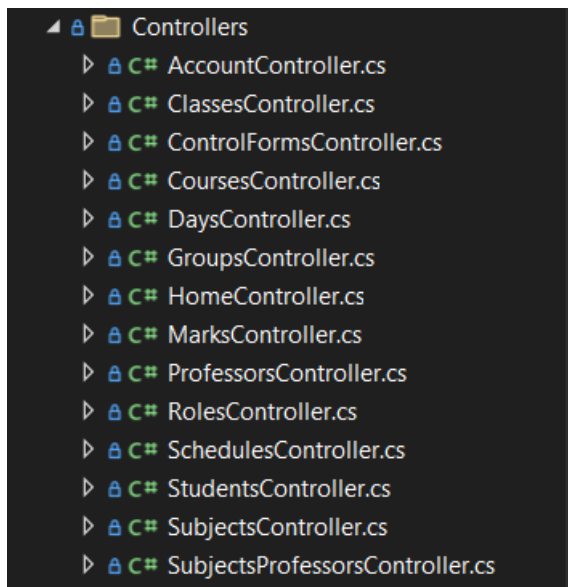


Рисунок 7 – Папка Controllers

Загалом, усі контролери та їх представлення відповідають таблицям бази даних. Виключенням є деякі класи, що було створено пізніше для інших цілей. До прикладу контролер RolesController.cs був створений для керування ролями зареєстрованих користувачів. А AccountController.cs відповідає саме за реєстрацію та авторизацію на платформі.

Відповідно, представлення цих контролерів відповідають за відображення цього функціоналу на сайті.

Для зручного пошуку необхідної інформації у розділах, було додано елементи фільтрації. Для цього було змінено метод Index() у відповідних контролерах (див. рис. 8).

```

public async Task<IActionResult> Index(string markControlForm, string searchString, string markSubject)
{
    if (_context.Marks == null)
    {
        return Problem("Entity set 'MvcMarkContext.Mark' is null.");
    }

    IQueryable<string> controlFormQuery = from m in _context.Marks
                                         orderby m.ControlForm
                                         select m.ControlForm.Name;

    var marks = from m in _context.Marks.Include(m => m.ControlForm).Include(m => m.Student).Include(m => m.Subject)
                select m;

    IQueryable<string> subjectQuery = from m in _context.Marks
                                     orderby m.Subject
                                     select m.Subject.Name;

    if (!string.IsNullOrEmpty(searchString))
    {
        marks = marks.Where(s => s.Student.FullName!.Contains(searchString)).Include(m => m.ControlForm).Include(m => m.Student).Include(m => m.Subject);
    }

    if (!string.IsNullOrEmpty(markControlForm))
    {
        marks = marks.Where(x => x.ControlForm.Name == markControlForm).Include(m => m.ControlForm).Include(m => m.Student).Include(m => m.Subject);
    }

    if (!string.IsNullOrEmpty(markSubject))
    {
        marks = marks.Where(x => x.Subject.Name == markSubject).Include(m => m.ControlForm).Include(m => m.Student).Include(m => m.Subject);
    }

    var marksFiltersVM = new MarkFiltersViewModel
    {
        ControlForms = new SelectList(await controlFormQuery.Distinct().ToListAsync()),
        Subjects = new SelectList(await subjectQuery.Distinct().ToListAsync()),
        Marks = await marks.ToListAsync()
    };

    return View(marksFiltersVM);
}

```

Рисунок 8 – Приклад зміненого методу Index() для фільтрації балів студентів

А також додано нові моделі у папку ViewModels (див. рис. 9).

```

1  using Microsoft.AspNetCore.Mvc.Rendering;
2  using UniversityPlatform.Models;
3
4  namespace UniversityPlatform.ViewModels
5  {
6      8 references
7      public class MarkFiltersViewModel
8      {
9          14 references
10         public List<Mark>? Marks { get; set; }
11         4 references
12         public SelectList? ControlForms { get; set; }
13
14         4 references
15         public SelectList? Subjects { get; set; }
16         2 references
17         public string? MarkControlForm { get; set; }
18         2 references
19         public string? MarkSubject { get; set; }
20         0 references
21         public string? SearchString { get; set; }
22     }
23 }

```

Рисунок 9 – Клас фільтрації балів студентів, доданий в папку ViewModels

Для відображення елементів фільтрації у представленнях, були прописані необхідні рядки (див. рис. 10).

```
<form asp-controller="Marks" asp-action="Index">
  <p>
    Предмет:
    <select asp-for="MarkSubject" asp-items="Model.Subjects">
      <option value="">Всі</option>
    </select>
    Форма контролю: <select asp-for="MarkControlForm" asp-items="Model.ControlForms">
      <option value="">Всі</option>
    </select>
    ПІБ: <input type="text" name="SearchString" />
    <input type="submit" value="Filter" />
  </p>
</form>
```

Рисунок 10 – Додані рядки у представлення балів для відображення фільтрів

Наступним кроком у розробці було формування та збереження інформації у форматі MS Excel. А саме: експорт (див. рис. 11) та імпорт файлів. Для цього була додана одна з бібліотек для взаємодії з файлами Excel -ClosedXML.

```
public ActionResult Export()
{
    using (XLWorkbook workbook = new XLWorkbook(XLEventTracking.Disabled))
    {
        var groups = _context.Groups.Include("Students").ToList();
        foreach (var g in groups)
        {
            var worksheet = workbook.Worksheets.Add(g.Name);

            worksheet.Cell("A1").Value = "ПІБ";
            worksheet.Cell("B1").Value = "Група";
            worksheet.Row(1).Style.Font.Bold = true;
            var students = g.Students.ToList();

            for (int i = 0; i < students.Count; i++)
            {
                worksheet.Cell(i + 2, 1).Value = students[i].FullName;
                worksheet.Cell(i + 2, 2).Value = students[i].Group.Name;
            }
        }
        using (var stream = new MemoryStream())
        {
            workbook.SaveAs(stream);
            stream.Flush();

            return new FileContentResult(stream.ToArray(),
                "application/vnd.openxmlformats-officedocument.spreadsheetml.sheet")
            {
                FileName = $"university_platform_{DateTime.UtcNow.ToShortDateString()}.xlsx"
            };
        }
    }
}
```

Рисунок 11 – Приклад методу для експорту файлів

Далі проводилась робота над автентифікацією, авторизацією та ролями. Для реалізації цього етапу використовувалась технологія ASP.NET Identity, яка представляє вбудовану в ASP.NET систему автентифікації і авторизації,

що надає змогу користувачам створювати облікові записи, автентифікуватися, керувати обліковими записами, або використовувати для входу на сайт облікові записи зовнішніх провайдерів.

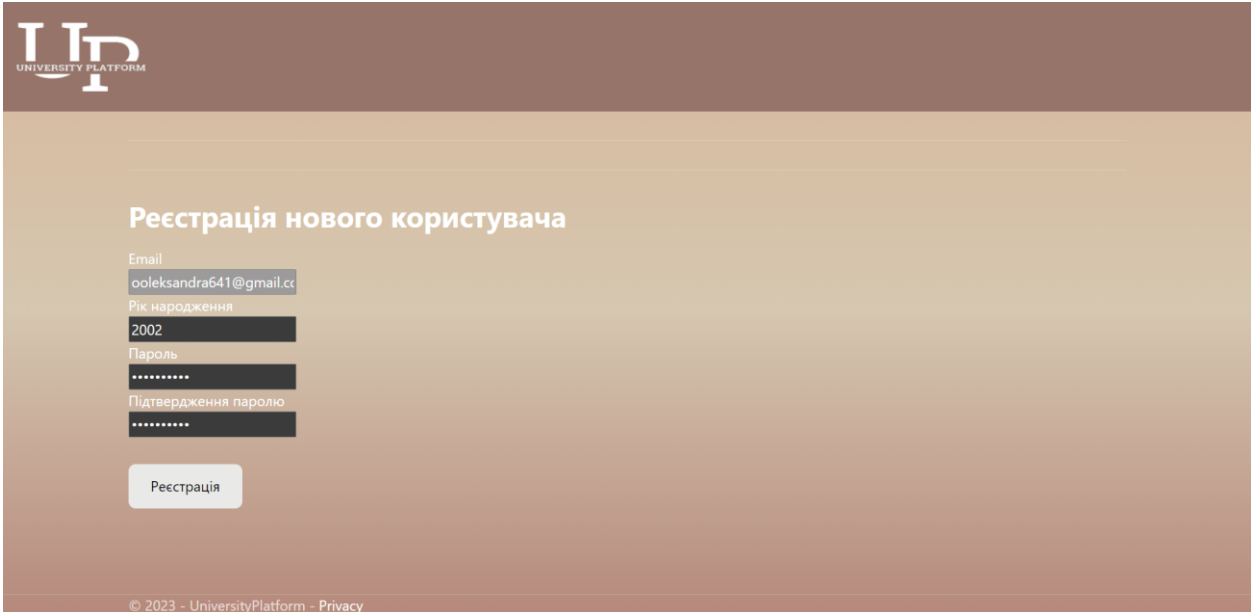
Заключним етапом у створенні даного вебзастосунку була робота над стилізацією та зміною зовнішнього вигляду програмного продукту, що був здійснений за допомогою фреймворку Bootstrap.

РОЗДІЛ 4. ІНСТРУКЦІЯ КОРИСТУВАЧА

Для наглядного прикладу використання платформи, нижче описано способи взаємодії для усіх типів ролей: студента, викладача, методиста та адміністратора.

Перед тим, як отримати доступ до усього функціоналу платформи, необхідно зареєструватись, або увійти (для попередньо зареєстрованого користувача) на сайт.

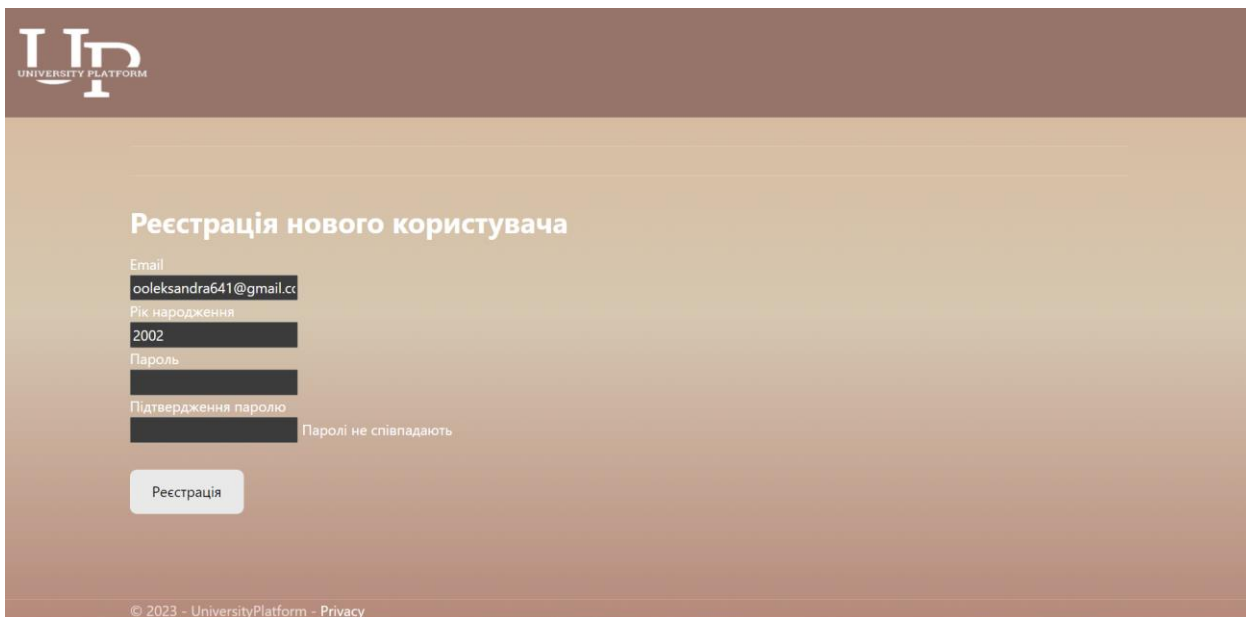
Для реєстрації користувачеві необхідно заповнити поля «Email», «Рік народження», «Пароль» та «Підтвердження паролю» (див. рис. 12).



The image shows a registration form on the University Platform. At the top left is the logo 'UP UNIVERSITY PLATFORM'. The main heading is 'Реєстрація нового користувача'. Below it are five input fields: 'Email' (containing 'ooleksandra641@gmail.cc'), 'Рік народження' (containing '2002'), 'Пароль' (masked with dots), and 'Підтвердження паролю' (masked with dots). A 'Реєстрація' button is at the bottom. A footer at the bottom left reads '© 2023 - UniversityPlatform - Privacy'.

Рисунок 12 – Реєстрація нового користувача

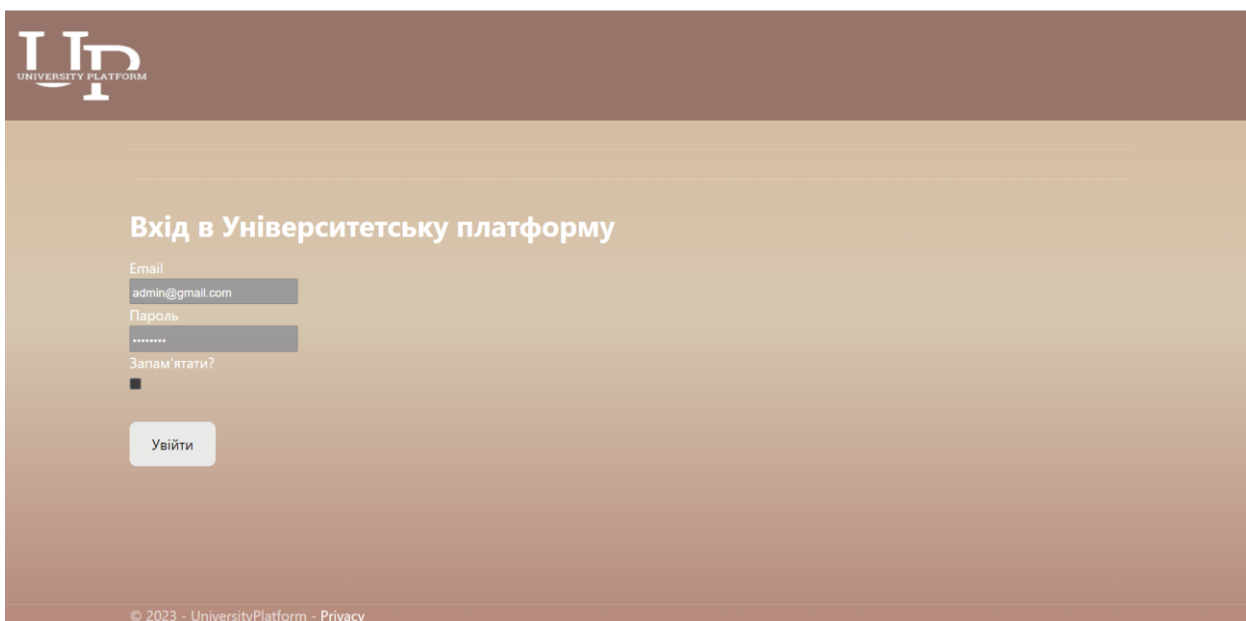
У разі введення некоректних даних, користувач побачить повідомлення про помилку, як зображено на рис. 13.



The screenshot shows the 'University Platform' logo at the top left. The main heading is 'Реєстрація нового користувача'. The form includes fields for 'Email' (filled with 'oleksandra641@gmail.com'), 'Рік народження' (filled with '2002'), 'Пароль', and 'Підтвердження паролю'. A red error message 'Паролі не співпадають' is displayed next to the password confirmation field. A 'Реєстрація' button is at the bottom of the form. The footer contains '© 2023 - UniversityPlatform - Privacy'.

Рисунок 13 – Приклад некоректного вводу даних

Для авторизації на платформу вимагається лише введення електронної пошти та паролю, що був вказаний під час реєстрації (див. рис. 14).



The screenshot shows the 'University Platform' logo at the top left. The main heading is 'Вхід в Університетську платформу'. The form includes fields for 'Email' (filled with 'admin@gmail.com') and 'Пароль' (filled with '.....'). There is a 'Запам'ятати?' checkbox which is checked. A 'Увійти' button is at the bottom of the form. The footer contains '© 2023 - UniversityPlatform - Privacy'.

Рисунок 14 – Вхід на платформу

Далі, в залежності від прав доступу, що наданий користувачеві відповідно до його ролей, платформа надає певні функціональні можливості.

4.1 Інструкція користувача в ролі студента

Увійшовши на платформу, студент бачить початкову сторінку, як показано на рис. 15. Зверху містяться розділи, на які він може перейти.

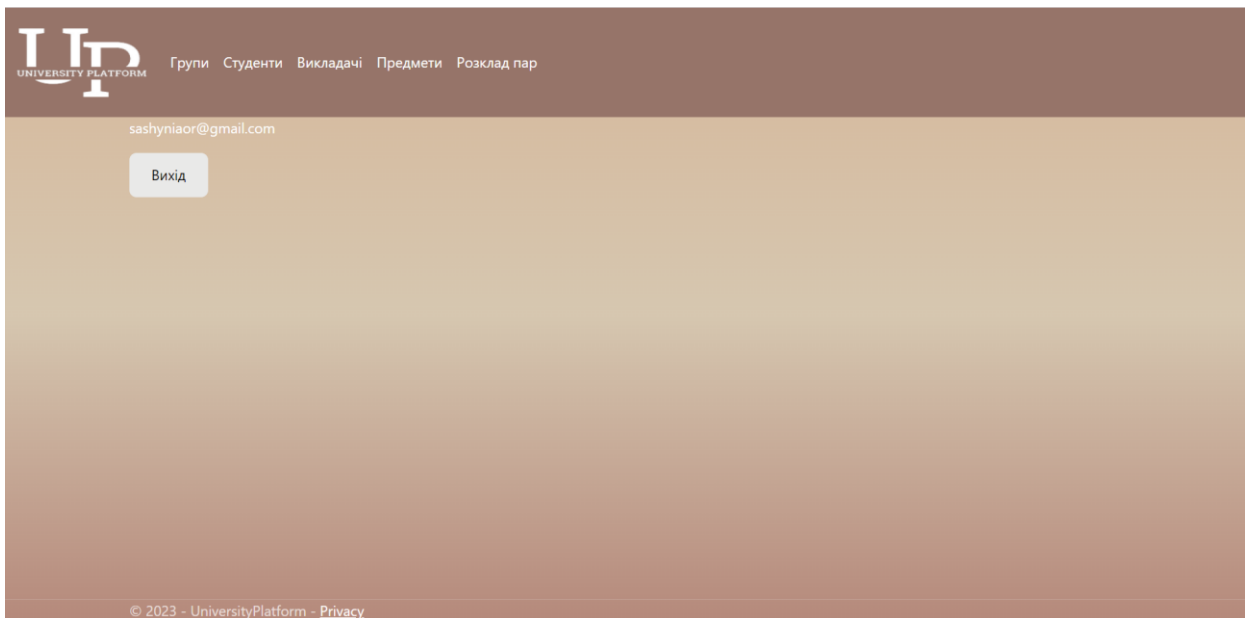


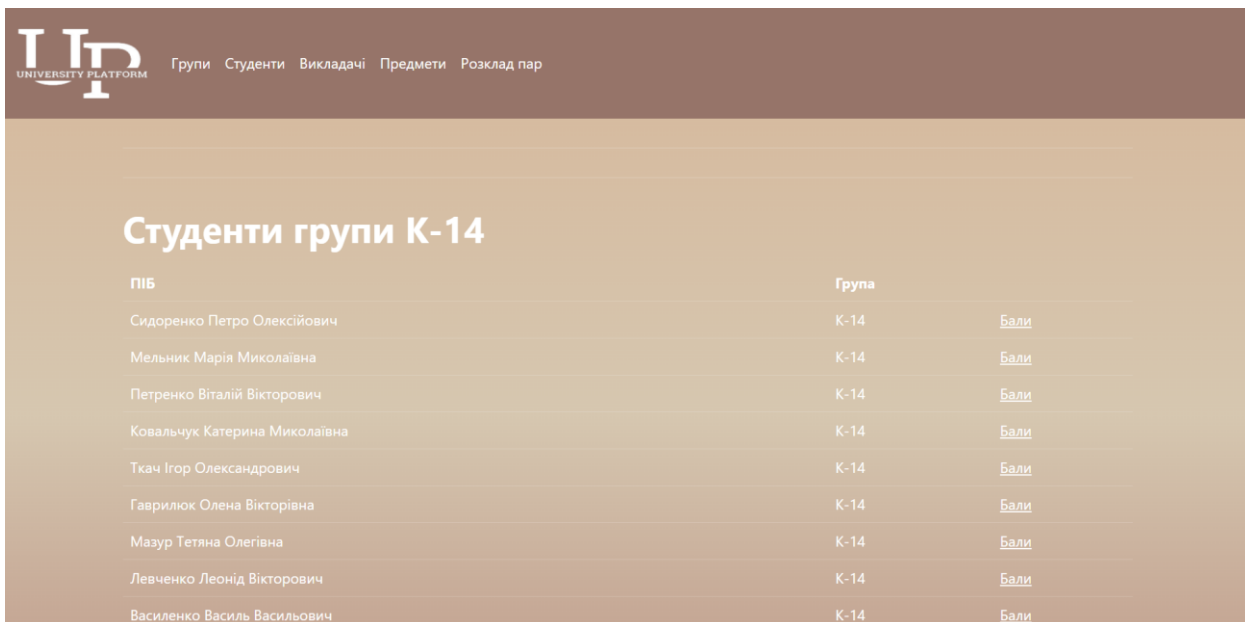
Рисунок 15 – Вигляд домашньої сторінки для студента

Перейшовши у розділ «Групи», студент має змогу переглянути увесь список груп (див. рис. 16).

Група	Курс	
К-14	1	Студенти Розклад
К-15	1	Студенти Розклад
К-16	1	Студенти Розклад
К-17	1	Студенти Розклад
К-24	2	Студенти Розклад
К-25	2	Студенти Розклад
К-26	2	Студенти Розклад
К-27	2	Студенти Розклад
ТК-31	3	Студенти Розклад

Рисунок 16 – Сторінка зі списком груп

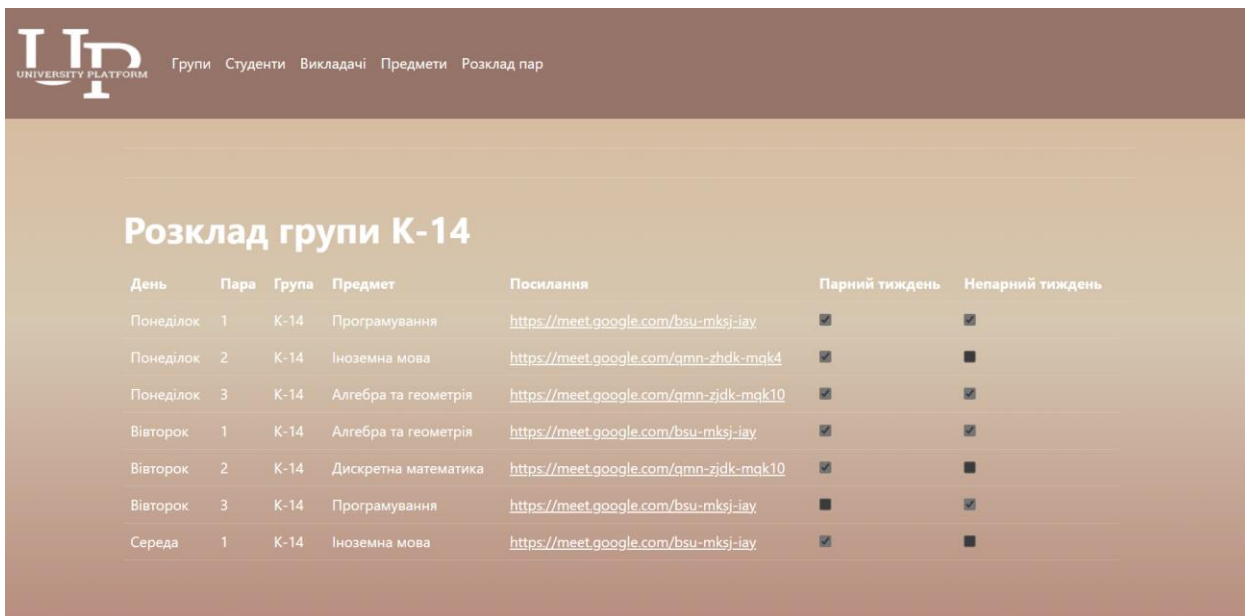
Також студент може переглянути усіх студентів групи натиснувши на «Студенти» (див. рис.17).



ПІБ	Група	Бали
Сидоренко Петро Олексійович	К-14	Бали
Мельник Марія Миколаївна	К-14	Бали
Петренко Віталій Вікторович	К-14	Бали
Ковальчук Катерина Миколаївна	К-14	Бали
Тхач Ігор Олександрович	К-14	Бали
Гаврилюк Олена Вікторівна	К-14	Бали
Мазур Тетяна Олегівна	К-14	Бали
Левченко Леонід Вікторович	К-14	Бали
Василенко Василь Васильович	К-14	Бали

Рисунок 17 – Студенти групи К-14

Щоб переглянути розклад для конкретної групи (див. рис. 18), необхідно натиснути на «Розклад» (див. рис. 16).



День	Пара	Група	Предмет	Посилання	Парний тиждень	Непарний тиждень
Понеділок	1	К-14	Програмування	https://meet.google.com/bsu-mksj-iaj	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Понеділок	2	К-14	Іноземна мова	https://meet.google.com/qmn-zhdk-mqk4	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Понеділок	3	К-14	Алгебра та геометрія	https://meet.google.com/qmn-zjdk-mqk10	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Вівторок	1	К-14	Алгебра та геометрія	https://meet.google.com/bsu-mksj-iaj	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Вівторок	2	К-14	Дискретна математика	https://meet.google.com/qmn-zjdk-mqk10	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Вівторок	3	К-14	Програмування	https://meet.google.com/bsu-mksj-iaj	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Середа	1	К-14	Іноземна мова	https://meet.google.com/bsu-mksj-iaj	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Рисунок 18 –Розклад групи К-14

Розділ «Студенти» (див. рис. 19) містить перелік усіх студентів університету.

UNIVERSITY PLATFORM Курси Групи Предмети Студенти Викладачі Студентська успішність Розклади Розклад пар

Список студентів

Додати студента

ПІБ:

ПІБ	Група	
Сидоренко Петро Олексійович	К-14	Редагувати Бали Видалити
Мельник Марія Миколаївна	К-14	Редагувати Бали Видалити
Петренко Віталій Вікторович	К-14	Редагувати Бали Видалити
Ковальчук Катерина Миколаївна	К-14	Редагувати Бали Видалити
Ткач Ігор Олександрович	К-14	Редагувати Бали Видалити
Гаврилюк Олена Вікторівна	К-14	Редагувати Бали Видалити
Мазур Тетяна Олегівна	К-14	Редагувати Бали Видалити

Рисунок 19 – Розділ «Студенти»

Для того, щоб полегшити пошук конкретного студента, було додано фільтр, в якому для пошуку потрібно ввести ПІБ студента (іноді вистачає лише Прізвища, як показано на рис. 20).

UNIVERSITY PLATFORM Курси Групи Предмети Студенти Викладачі Студентська успішність Розклади Розклад пар

Список студентів

Додати студента

ПІБ:

ПІБ	Група	
Сидоренко Петро Олексійович	К-14	Редагувати Бали Видалити
Мельник Марія Миколаївна	К-14	Редагувати Бали Видалити
Петренко Віталій Вікторович	К-14	Редагувати Бали Видалити
Ковальчук Катерина Миколаївна	К-14	Редагувати Бали Видалити
Ткач Ігор Олександрович	К-14	Редагувати Бали Видалити
Гаврилюк Олена Вікторівна	К-14	Редагувати Бали Видалити
Мазур Тетяна Олегівна	К-14	Редагувати Бали Видалити

Рисунок 20 – Приклад застосування фільтрів на сторінці «Студенти»

Для отримання результату (див. рис. 21), необхідно натиснути кнопку «Фільтрувати».

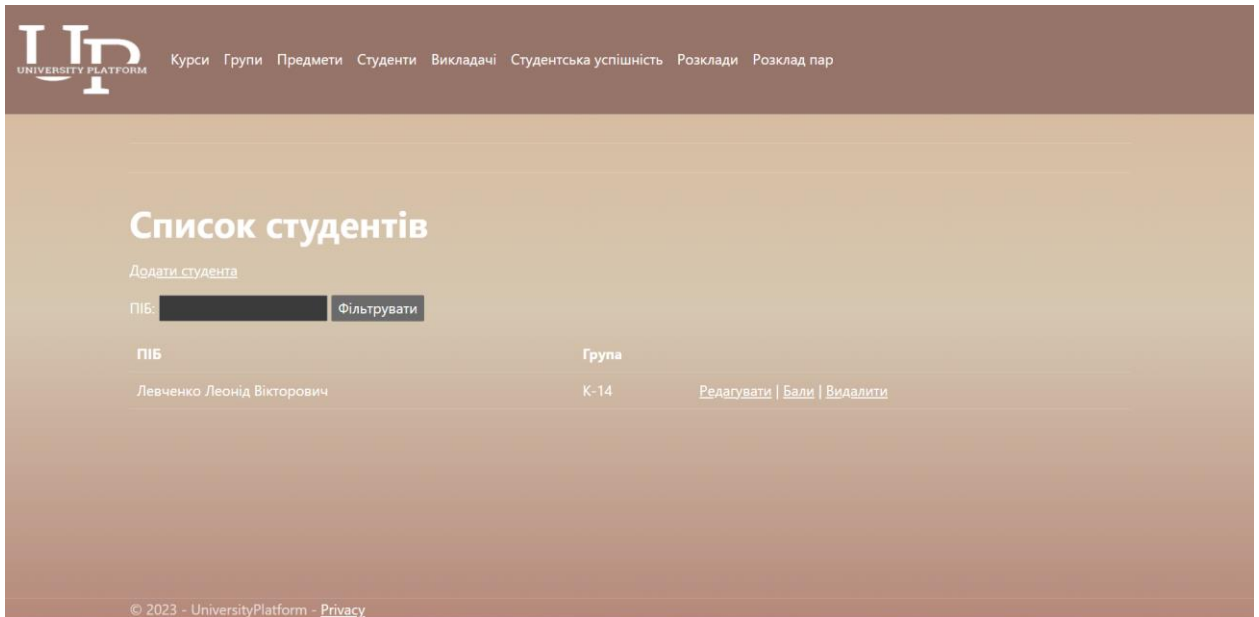


Рисунок 21 – Відфільтрований список студентів за ПІБ

Для кожного студента є можливість проглянути його бали, натиснувши на «Бали» на сторінці зі списком студентів. Сторінка зі списком отриманих балів обраного студента зображена на рис. 22.

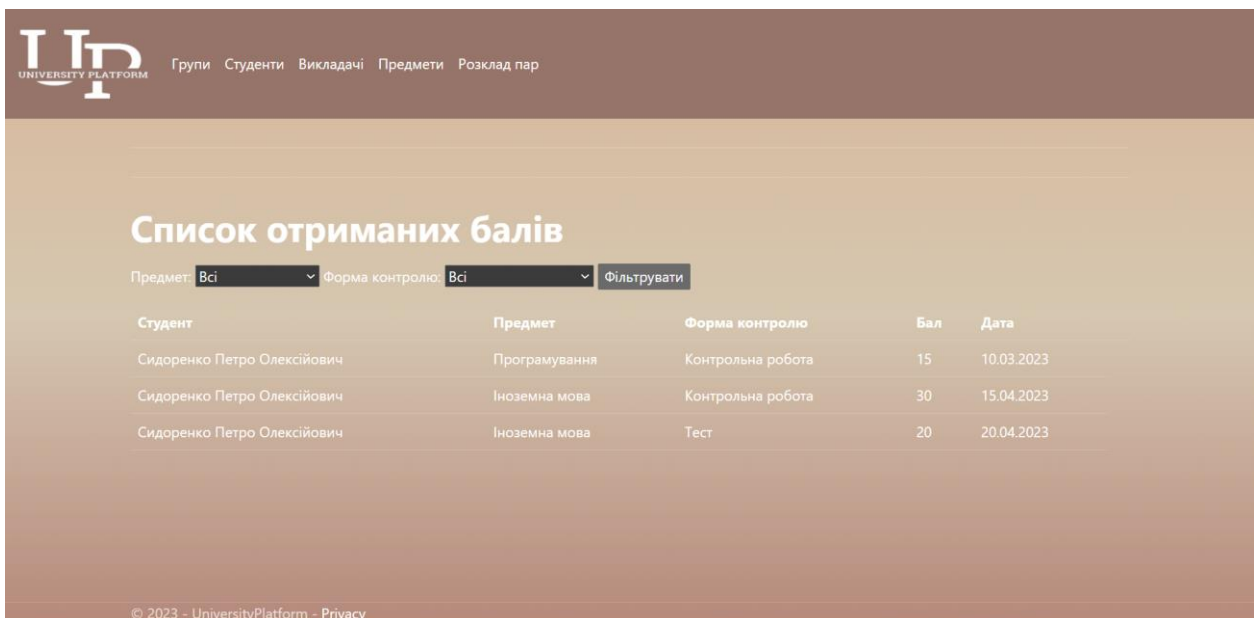


Рисунок 22 – Список отриманих балів конкретного студента

Розділ «Предмети» містить перелік усіх предметів, що викладаються в університеті (див. рис. 23).

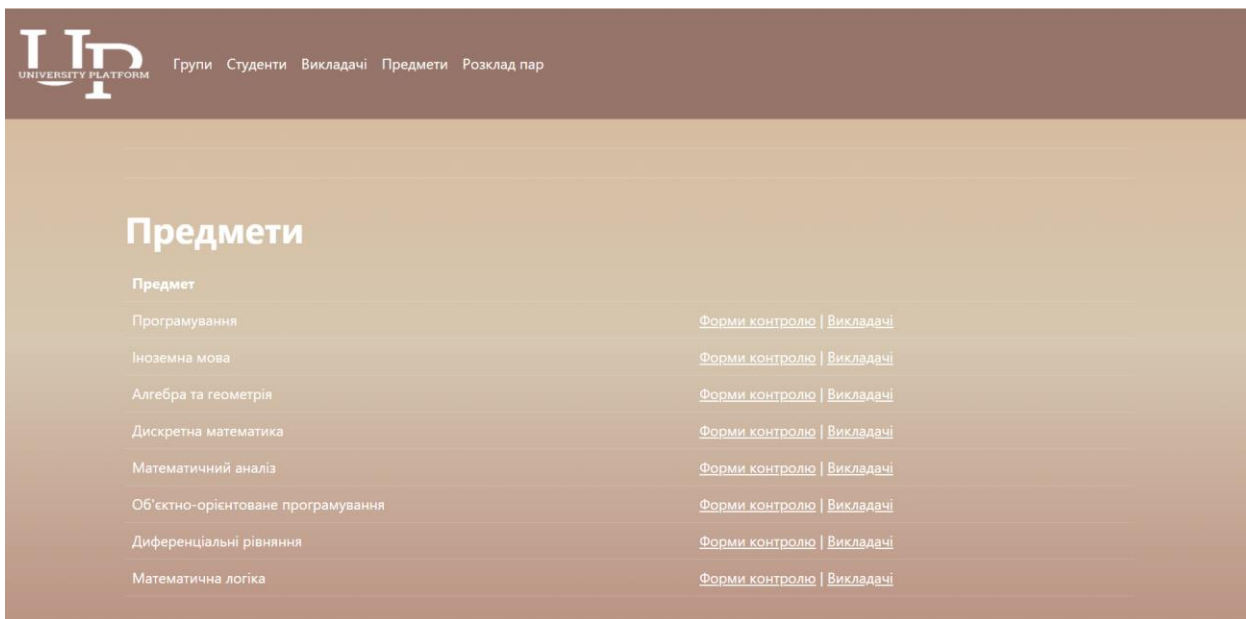


Рисунок 23 – Сторінка розділу «Предмети»

Розділ «Розклад пар» слугує нагадуванням для студентів о котрій годині починаються та закінчуються пари (див. рис. 24).

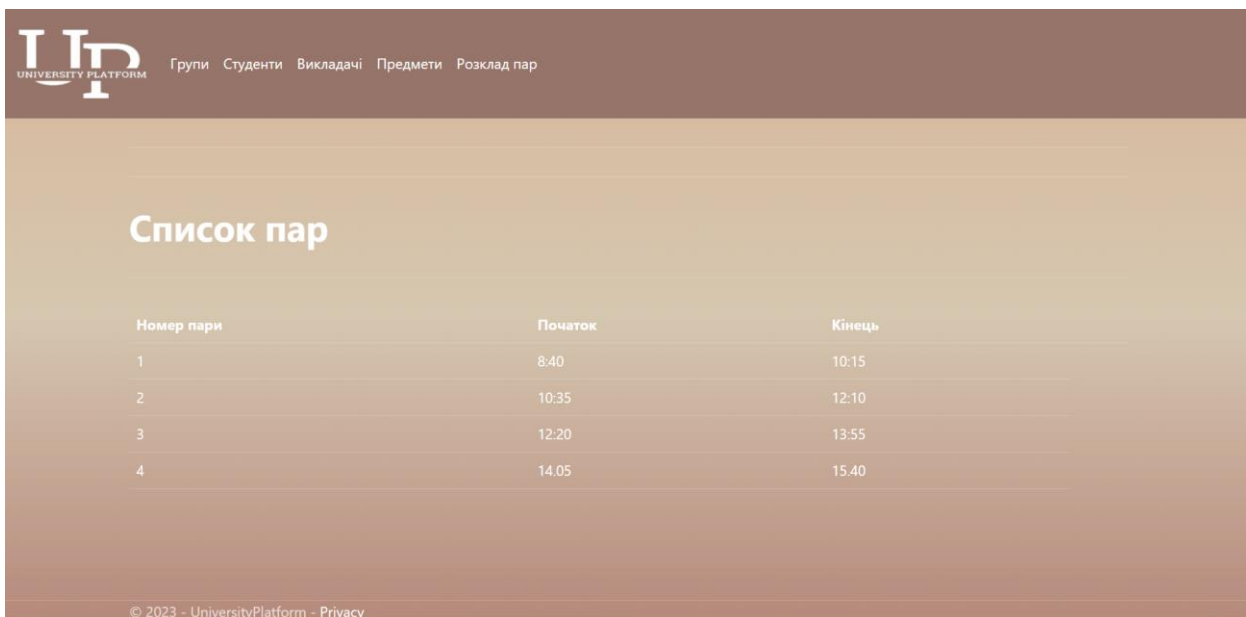
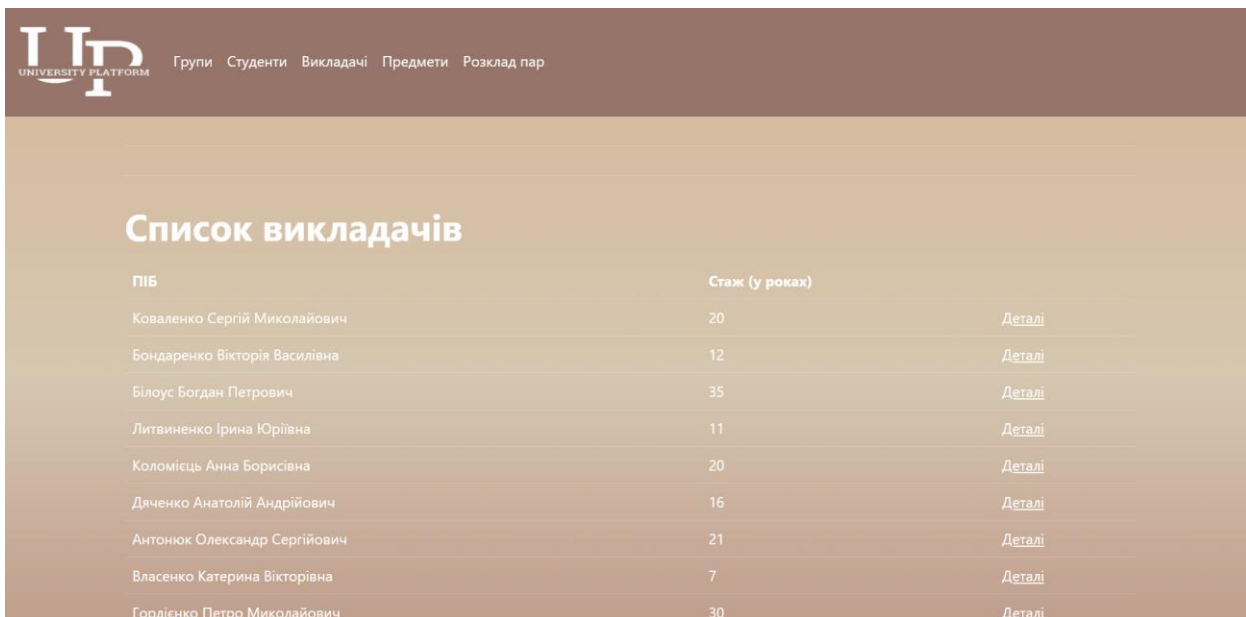


Рисунок 24 – Сторінка розділу «Розклад пар»

У розділі «Викладачі» міститься інформація про викладачів, які працюють в університеті (див. рис. 25)



ПІБ	Стаж (у роках)	
Коваленко Сергій Миколайович	20	Деталі
Бондаренко Вікторія Василівна	12	Деталі
Білоус Богдан Петрович	35	Деталі
Литвиненко Ірина Юріївна	11	Деталі
Коломієць Анна Борисівна	20	Деталі
Дяченко Анатолій Андрійович	16	Деталі
Антонюк Олександр Сергійович	21	Деталі
Власенко Катерина Вікторівна	7	Деталі
Гордієнко Петро Миколайович	30	Деталі

Рисунок 25 – Сторінка розділу «Викладачі»

4.2 Інструкція користувача в ролі викладача

Після авторизації на платформі, викладач переходить на домашню сторінку (див. рис 26).

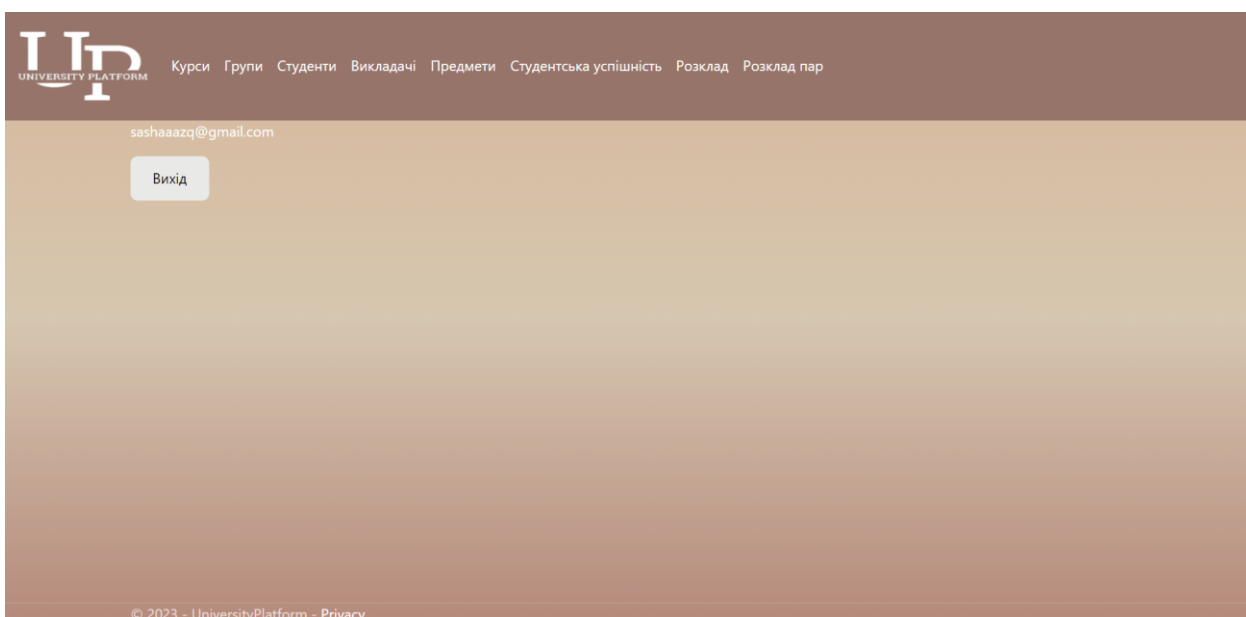


Рисунок 26 – Вигляд домашньої сторінки для викладача

В цілому, функціонал викладача схожий до функціоналу студента, втім він має більше можливостей і доступних розділів. Одним з таких розділів є розділ «Курси», що допомагає знайти необхідну групу за курсом (див. рис. 27).

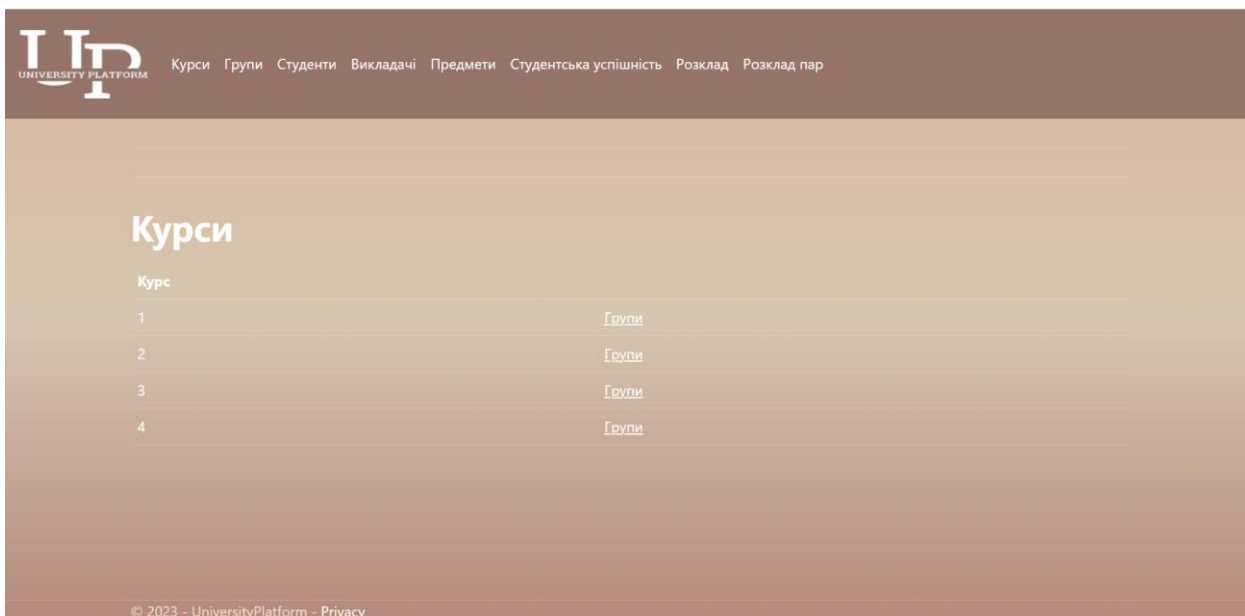


Рисунок 27 – Сторінка розділу «Курси»

За розділом «Студентська успішність», викладач може переглянути список балів потрібних йому студентів, використовуючи доступні фільтри на сторінці (див. рис. 28).

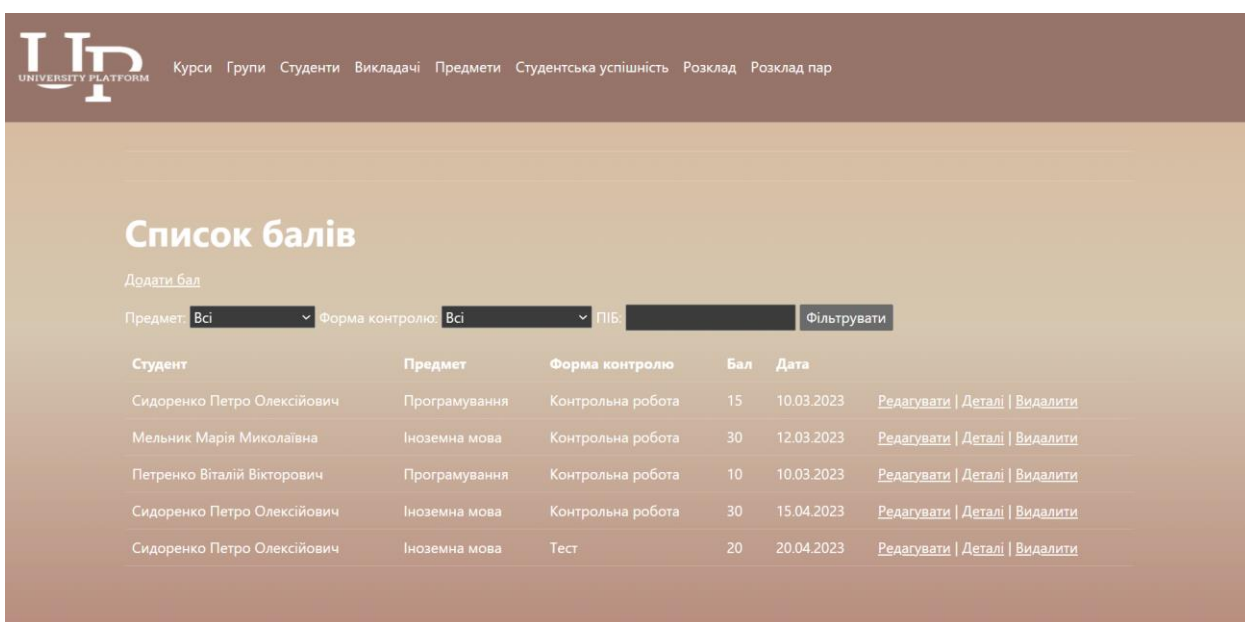
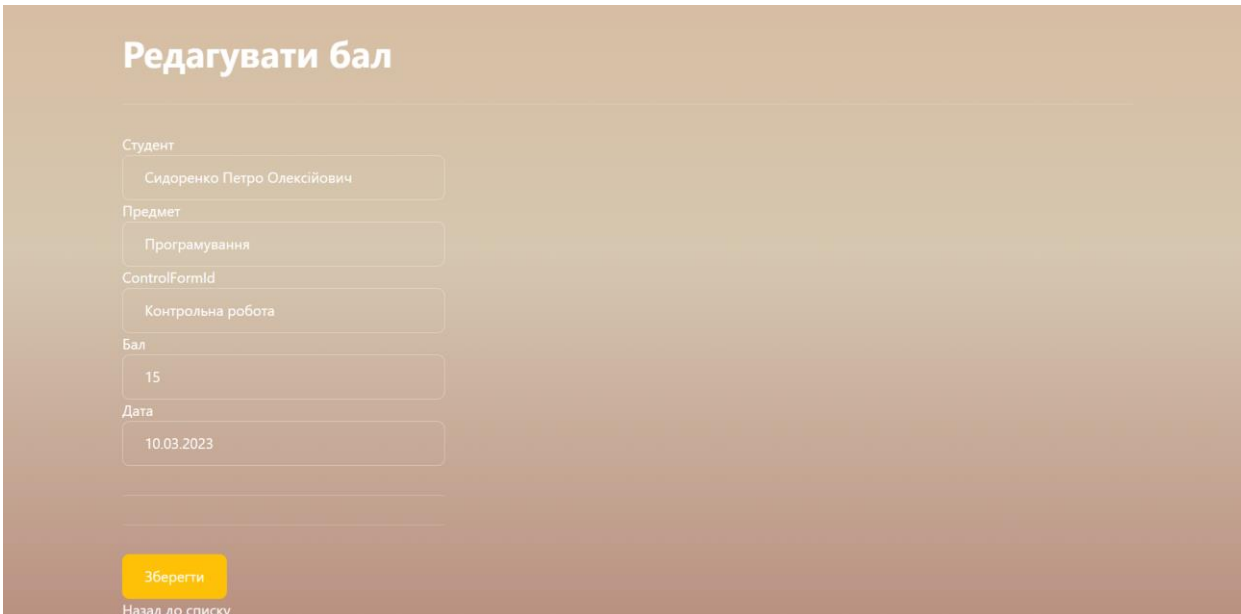


Рисунок 28 – Сторінка розділу «Студентська успішність»

А також додати новий бал, редагувати і видалити існуючі за необхідності. Приклади редагування можна переглянути на рис. 29.



Редагувати бал

Студент
Сидоренко Петро Олексійович

Предмет
Програмування

ControlFormId
Контрольна робота

Бал
15

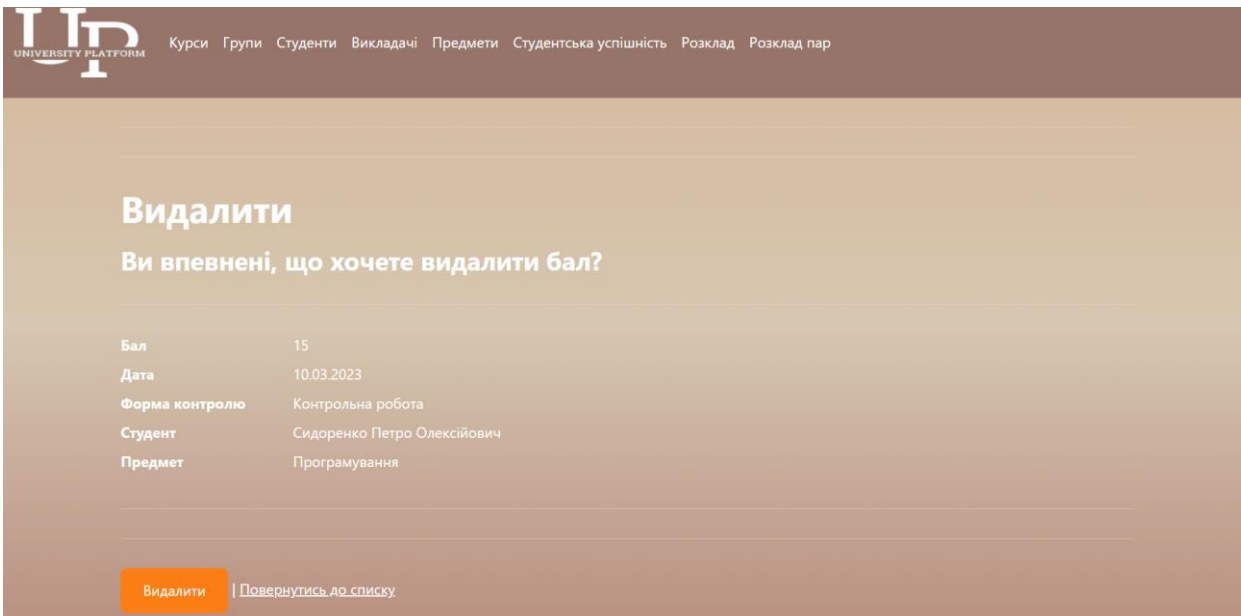
Дата
10.03.2023

[Зберегти](#)

[Назад до списку](#)

Рисунок 29 – Редагування балу

Приклад видалення наведено на рис. 30.



Видалити

Ви впевнені, що хочете видалити бал?

Бал	15
Дата	10.03.2023
Форма контролю	Контрольна робота
Студент	Сидоренко Петро Олексійович
Предмет	Програмування

[Видалити](#) | [Повернутись до списку](#)

Рисунок 30 – Видалення балу

У розділі «Групи» для викладача додано можливість завантажувати списки груп у файли для подальшого збереження на своїх локальних пристроях (див. рис. 31).

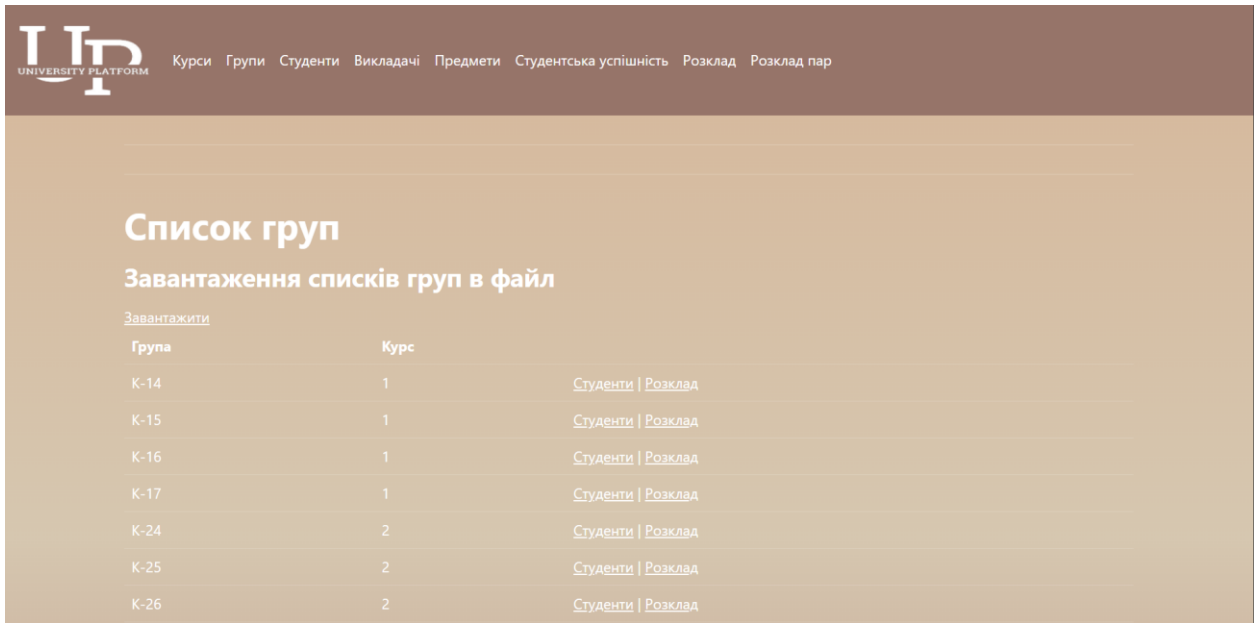


Рисунок 31 – Сторінка розділу «Групи» з можливістю завантажувати списки груп у файли

Також, до основних функціональних можливостей для викладача є змога створювати, редагувати та видаляти форми контролю для предметів, перейшовши на сторінку «Форми контролю» (див. рис. 32) через розділ «Предмети» (див. рис. 23).

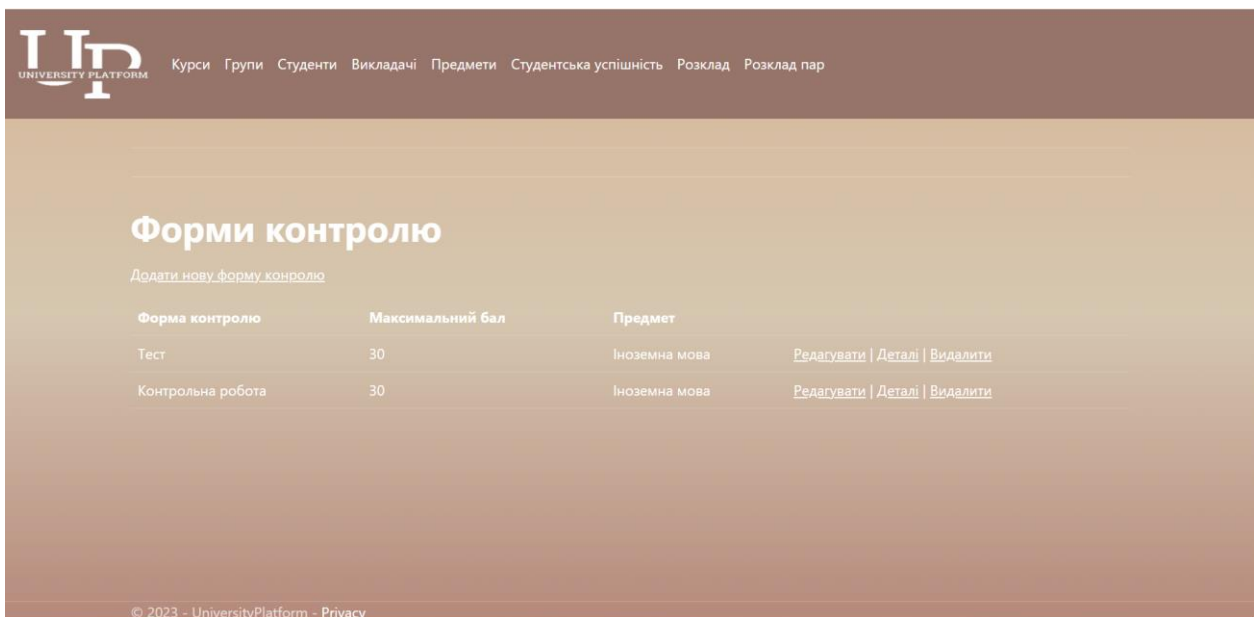
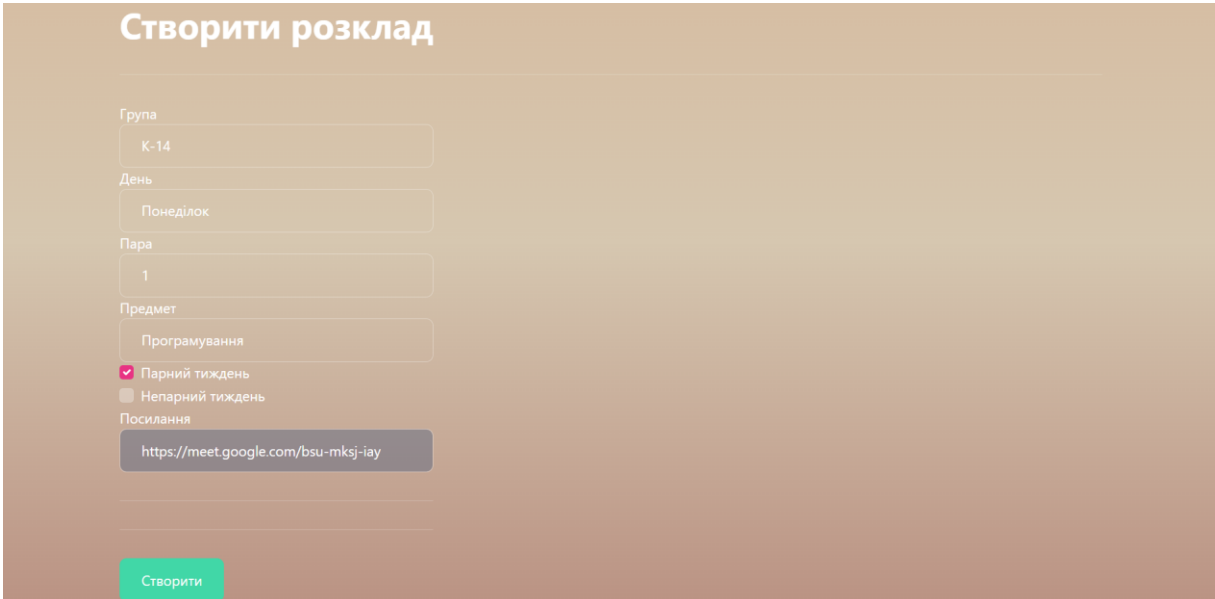


Рисунок 32 – Сторінка «Форми контролю» для предмету «Іноземна мова»

4.3 Інструкція користувача в ролі методиста

Для методиста наданий найбільший спектр функціональних можливостей, зокрема це змога переглядати, додавати (див. рис. 33), видаляти та редагувати усі наявні розділи, крім розділу студентської успішності та форм контролю (ці можливості надані лише користувачам у ролі викладача).



Створити розклад

Група
К-14

День
Понеділок

Пара
1

Предмет
Програмування

Парний тиждень
 Непарний тиждень

Посилання
<https://meet.google.com/bsu-mksj-iaay>

Створити

Рисунок 33 – Приклад додавання розкладу

З особливих можливостей для методиста також варто зазначити про здатність імпортувати Excel-файли для завантаження готових списків груп на платформу (див. рис. 34).

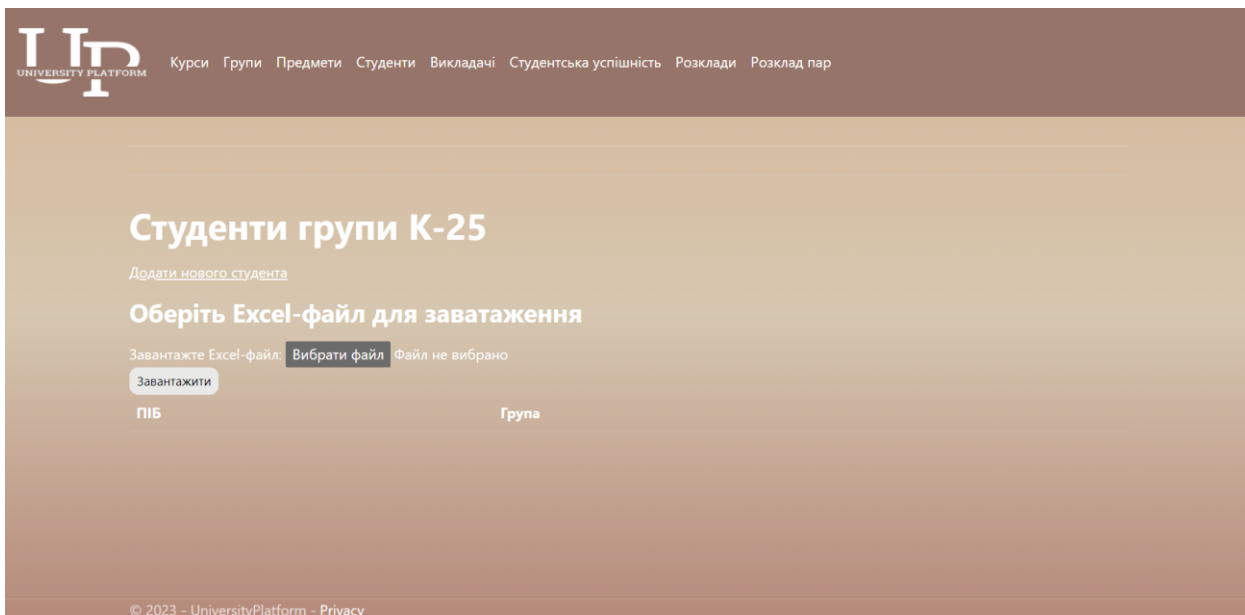


Рисунок 34 – Сторінка з можливістю імпортувати Excel-файли для завантаження готових списків груп на платформу

4.3 Інструкція користувача в ролі адміністратора

Адміністратору доступний розділ «Управління ролями», де знаходиться список можливих ролей (див. рис 35).

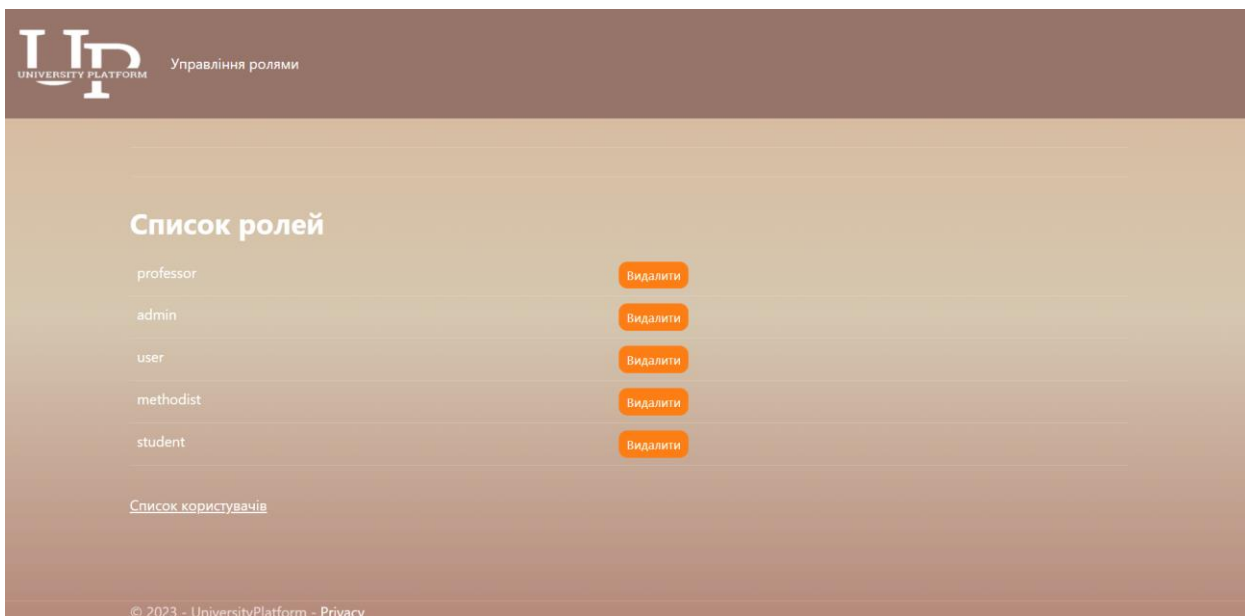


Рисунок 35 – Сторінка розділу «Управління ролями»

Звідси адміністратор може перейти на список користувачів платформи, де може надати права доступу конкретному користувачу (див. рис. 36).

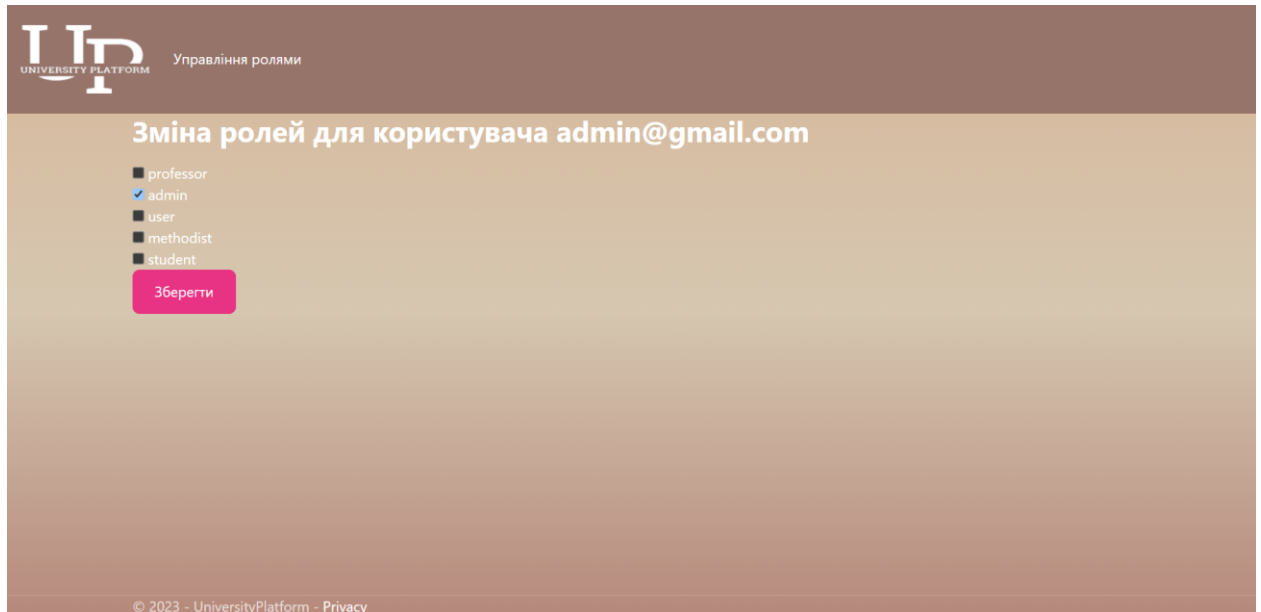


Рисунок 36 – Зміна ролей для користувача

ВИСНОВКИ

Під час написання кваліфікаційної роботи було розроблено вебзастосунок для підтримки організації освітнього процесу з використанням технологій дистанційного навчання в ЗВО. При цьому виконано наступні завдання: проаналізовано предметну область, спроектовано застосунок, розроблено діаграми прецедентів для усіх видів ролей та діаграму бази даних, створено базу даних в SQL Server Management Studio (SSMS), реалізовано формування та збереження звітів у форматі Excel (експорт та імпорт файлів) та забезпечено підтримку автентифікації, авторизації та керування ролями.

Як результат – отримали працюючий програмний продукт, який є зручним рішенням для підтримки і організації навчального процесу в ЗВО, зокрема для університетів. Такий вебзастосунок може спростити роботу для методистів університету, дозволяючи швидко вносити зміни в розклад, завантажувати файли і маніпулювати даними в інший спосіб на платформі, завдяки автоматизації усіх перелічених процесів.

Для студентів така платформа являється надзвичайно зручним джерелом для збору і аналізу необхідної для них інформації, що стосується освітніх процесів. А для викладачів вебзастосунок допомагає не лише у зборі інформації, а й являється засобом для контролю успішності студентів.

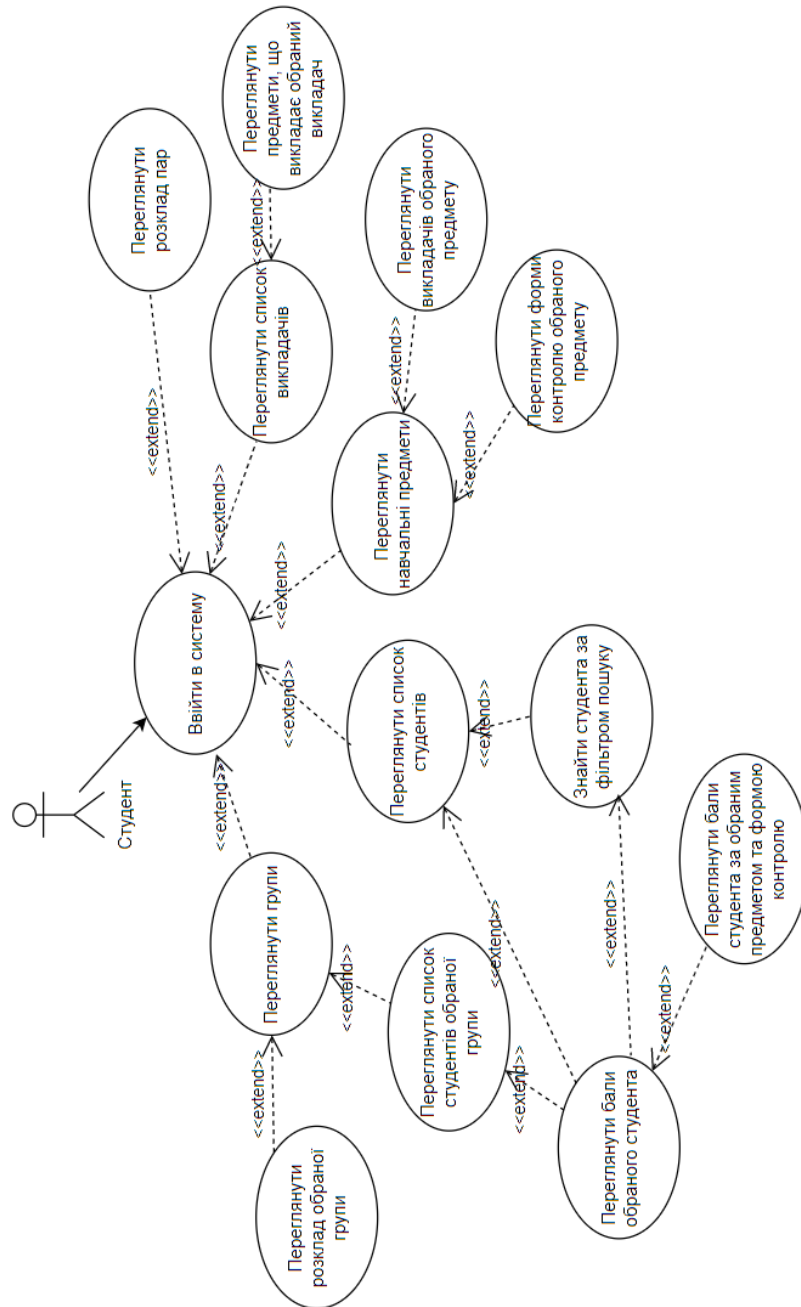
Завдяки роботі над кваліфікаційною роботою було покращено і закріплено навички, необхідні для розробки подібних вебзастосунків, а також розроблено навчальну платформу для автоматизації процесу організації освітніх процесів у ЗВО з використанням технологій дистанційного навчання.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Автоматизована інформаційна система управління закладом, М.М. Косіюк, К.Е. Більовський, В.М. Лисак, 2023.
2. Закон України «Про вищу освіту» [Електронний ресурс]. – Режим доступу: <https://zakon.rada.gov.ua/laws/show/1556-18#Text>
3. Moodle - Open-source learning platform [Електронний ресурс]. – Режим доступу: <https://moodle.org/?lang=uk>
4. Triton Student [Електронний ресурс]. – Режим доступу: <https://student.triton.knu.ua/>
5. Автоматизована система управління «Вищий навчальний заклад» [Електронний ресурс] – Режим доступу до ресурсу: <https://vuz.osvita.net/>
6. АСУ «Вищий навчальний заклад» [Електронний ресурс]. – Режим доступу: https://vuz.osvita.net/wp-content/uploads/2021/09/ASU_VNZ.pdf
7. Entity Framework Core [Електронний ресурс]. – Режим доступу: <https://learn.microsoft.com/en-us/ef/core/>
8. Mark J.Price. C# 10 and .NET 6 – Modern Cross-Platform Development: Build apps, websites, and services with ASP.NET Core 6, Blazor, and EF Core 6 using Visual Studio 2022 and Visual Studio Code, 6th Edition, 2021. 826 с.
9. Get started with ASP.NET Core MVC [Електронний ресурс]. – Режим доступу: <https://learn.microsoft.com/en-us/aspnet/core/tutorials/first-mvc-app/start-mvc?view=aspnetcore-7.0&tabs=visual-studio>
10. Bootstrap [Електронний ресурс]. – Режим доступу: <https://getbootstrap.com/>
11. MVC Design Pattern [Електронний ресурс]. – Режим доступу: <https://www.geeksforgeeks.org/mvc-design-pattern/>

ДОДАТКИ

Додаток А. Use case діаграма студента



Додаток Г. Use case діаграма адміністратора

