МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
УМАНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ САДІВНИЦТВА
Факультет економіки і підприємництва
Кафедра інформаційних технологій

Допущено до захисту
"_____" _____ 2023 р.
Зав. кафедри, к.т.н., доцент
_____ **Ліщук Р.І.**

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ ОБЛІКУ ТА КОНТРОЛЮ ВИХОВНОЇ РОБОТИ ЗІ СТУДЕНТСЬКИМИ ГРУПАМИ

Виконав – студент 2 курсу, групи 21м-кн
освітньої програми "Комп’ютерні науки"
спеціальності 122 Комп’ютерні науки
Алексеєнко Антон Андрійович

Науковий керівник – к.т.н., доцент
Ліщук Роман Ігорович

Умань – 2023

Зміст

ВСТУП 4

РОЗДІЛ 1 АНАЛІЗ ПРОБЛЕМИ ТА ПОСТАНОВКА ЗАДАЧІ 6

1.1. Актуальність роботи 6

1.2. Аналіз існуючих систем 7

1.3. Інтеграція із існуючою АСУ 10

1.4. Постановка задачі 12

РОЗДІЛ 2 ВИБІР ТЕХНОЛОГІЙ ТА АРХІТЕКТУРИ СИСТЕМИ 14

2.1. Вибір технологій розробки 14

2.2. Архітектура системи 24

2.3. Проектування бази даних 29

2.4. Забезпечення об'єктивності та достовірності даних 31

РОЗДІЛ 3 РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ ОБЛІКУ ТА КОНТРОЛЮ ВИХОВНОЇ РОБОТИ З СТУДЕНТСЬКИМИ ГРУПАМИ 34

3.1. Розробка бази даних 34

3.2. Розробка користувацького інтерфейсу для користувача із роллю куратора 41

3.3. Розробка користувацького інтерфейсу адміністратора 51

3.4. Реалізація функціоналу збору та аналізу звітної інформації 54

ВИСНОВКИ 60

СПИСОК ЛІТЕРАТУРИ 61

ДОДАТОК А ЛІСТИНГ КОДУ ІНФОРМАЦІЙНОЇ СИСТЕМИ 66

ДОДАТОК Б ЛІСТИНГ МОДУЛЯ ІНТЕГРАЦІЇ ІЗ АСУ 78

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ І ТЕРМІНІВ

ІТ – інформаційні технології

ІС – інформаційна система

АСУ – автоматизована система управління університетом

БД – база даних

СУБД – система управління базами даних

SQL – Structured Query Language

HTTP – HyperText Transfer Protocol

UML – Unified Modeling Language

ISO – International Organization for Standardization

HTML – HyperText Markup Language

XML – Extensible Markup Language

PHP – Hypertext Preprocessor

CSV – Comma-Separated Values

CSS – Cascading Style Sheets

VS Code – Microsoft Visual Studio Code

ВСТУП

В умовах швидкого розвитку інформаційних технологій, університети по всьому світу активно впроваджують системи для автоматизації різних аспектів навчального процесу. Однак, облік та проведення виховної роботи зі студентськими групами залишається менш автоматизованим аспектом, що вимагає спеціалізованих рішень.

В умовах сучасного освітнього процесу, виховна робота зі студентами набуває особливої актуальності. Це не лише допомагає формувати професійні та особистісні якості майбутніх фахівців, але й є ключовим фактором для створення сприятливого навчального та соціального середовища. Однак, в умовах великого обсягу адміністративних завдань та зростаючих вимог до якості освіти, ефективний облік та управління виховною роботою стає все більш складним завданням.

Відсутність автоматизованих систем для обліку виховної роботи може призвести до низки проблем: від затрат часу на рутинні завдання до втрати важливої інформації та нездатності провести якісний аналіз ефективності виховних заходів. Це, в свою чергу, може негативно позначитися на якості виховної роботи та, як наслідок, на загальному рівні освіти.

В Україні та інших країнах з розвиненою системою вищої освіти існує ряд програмних продуктів, спрямованих на автоматизацію різних аспектів управління освітнім процесом. Однак, не всі з них забезпечують комплексний підхід до вирішення проблеми обліку та контролю виховною роботою, що ставить питання про необхідність розробки спеціалізованих інформаційних систем для цієї мети.

Таким чином, сучасний стан проблеми обліку та управління виховною роботою в університетах вимагає нових підходів та технологічних рішень, які б забезпечили ефективність та якість даного процесу.

Об'єктом дослідження є процеси обліку та проведення виховної роботи зі студентськими групами в університетах, зокрема в Уманському національному університеті садівництва.

Предметом дослідження є розробка інформаційної системи, яка дозволить автоматизувати зазначені процеси, забезпечуючи їх ефективність, прозорість та надійність.

Метою дослідження є аналіз сучасного стану проблеми, визначення ключових вимог до системи, розробка та впровадження інформаційної системи для обліку та контролю виховної роботи зі студентськими групами, що відповідає сучасним стандартам та потребам університету.

Автоматизація виховної роботи є важливою та актуальною задачею, оскільки вона сприяє ефективному управлінню ресурсами, забезпечує точний облік даних та дозволяє відстежувати динаміку розвитку студентів.

РОЗДІЛ 1 АНАЛІЗ ПРОБЛЕМИ ТА ПОСТАНОВКА ЗАДАЧІ

1.1. Актуальність роботи

Двадцять перше століття – це час розвинених інформаційно-комунікаційних технологій, глобальної модернізації виробництва та розвитку людського капіталу. Державна політика України відіграє важливу роль у забезпеченні розвитку людського капіталу та досягненні економічних переваг у вигляді сталого зростання та конкурентоспроможної економіки, що означає суспільний та індивідуальний добробут, майбутній добробут та якість життя. та у сфері глобальної освіти та науки [1]. Зміна сучасного суспільного життя, потреби і запити ринку праці призвели до необхідності модифікації розвитку світового освітнього простору загалом і освітнього простору країни зокрема.

У сучасному освітньому процесі, де технології та інновації стають ключовими елементами навчального процесу, розробка інформаційних систем, які спрямовані на оптимізацію та автоматизацію роботи кураторів із студентськими групами, набуває особливої актуальності. Системи, які дозволяють кураторам ефективно вести облік, планувати та аналізувати виховну роботу, можуть значно полегшити їхню роботу, забезпечуючи при цьому високий рівень контролю та аналітики.

В умовах постійної зміни освітнього процесу та зростання вимог до якості навчання і виховання студентів, інформаційні системи стають невід'ємною частиною управлінського процесу в університетах. [2] Вони допомагають кураторам та адміністрації університету забезпечити системність виховної роботи, а також дозволяють відстежувати динаміку розвитку студентських груп.

Особливо актуальним питання розробки таких систем стає у контексті дистанційної освіти та гібридних навчальних форматів [3], де фізичний контакт між куратором та студентами обмежений, а потреба в ефективних інструментах для ведення обліку та контролю виховної роботи зі студентами зростає.

Дистанційна форма навчання може надати учням найшвидший доступ до нетрадиційних для класичної школи джерел інформації. Дистанційне навчання також надає змогу активізувати творчу діяльність школярів та підвищити результативність їх самостійної роботи, а вчителі, своєю чергою, можуть використовувати у своїй роботі абсолютно нові і беззаперечно ефективні методи навчання із застосуванням концептуального та математичного моделювання явищ та процесів [4].

Розробка інформаційної системи для обліку та контролю виховної роботи з студентськими групами відповідає актуальним потребам освітньої галузі, спрямована на підвищення ефективності роботи кураторів та оптимізацію процесів управління виховною роботою в університетах. Враховуючи це, дана робота є актуальною та важливою для сучасної освітньої практики.

1.2. Аналіз існуючих систем

В сучасному світі існує велика кількість систем, які дозволяють автоматизувати різні аспекти управління навчальним процесом, від академічного обліку до взаємодії між учасниками освітнього процесу. Однак, не всі з них здатні ефективно вирішувати задачі, пов'язані з виховною роботою, яка є невід'ємною частиною загальної системи освіти. Виховна робота вимагає специфічного підходу та інструментів для ефективного контролю та обліку. [5] В рамках даного роботи було вивчено декілька ключових систем, щоб зрозуміти, які можливості вони пропонують, які їхні переваги та обмеження. Це допоможе визначити, які аспекти потрібно врахувати при розробці нової системи для автоматизації обліку та проведення виховної роботи зі студентськими групами. Розгляд цих систем дозволить нам зробити вибір на користь найбільш оптимальних рішень, які можна буде впровадити в інформаційній системі. Після аналізу ринку наявних інформаційних систем для обліку та контролю виховної роботи зі студентськими групами, виокремлено

три таких для більш детального огляду та описано основні їх характеристики та функції: EduSys [5], SchoolSoft [7], Moodle [8].

EduSys є однією з найбільш популярних систем управління навчальним процесом. Ця система пропонує широкий спектр функціональних можливостей, включаючи академічний облік, планування розкладу, та взаємодію між студентами та викладачами. Однак, що стосується виховної роботи, EduSys має певні обмеження. Система фокусується переважно на академічних аспектах, і не надає спеціалізованих інструментів для обліку та управління виховним процесом. Наприклад, вона не дозволяє вести деталізований облік виховних заходів, планувати їх та аналізувати ефективність. Це може стати проблемою для викладачів, які ведуть виховну роботу і потребують більш гнучких та цілеспрямованих інструментів.

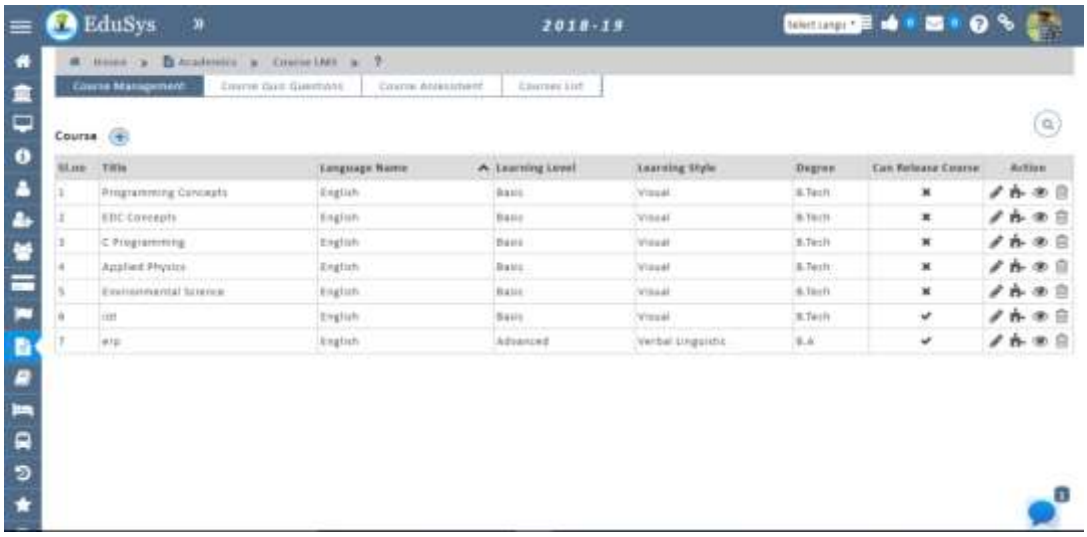


Рисунок 1.1 – Інформаційна система EduSys

Іншою системою управління навчальним процесом є SchoolSoft, яка включає в себе ряд функцій для академічного обліку та аналітики. Однією з її основних переваг є можливість аналізу даних для покращення навчального процесу. Однак, важливо зазначити, що система не фокусується на виховній роботі. Хоча SchoolSoft дозволяє збирати та аналізувати дані, її функціонал не розширюється на специфічні потреби виховної роботи, такі як планування та облік виховних заходів. У контексті нашої системи, де основна мета полягає в

автоматизації обліку та проведення виховної роботи, це є значущим обмеженням.

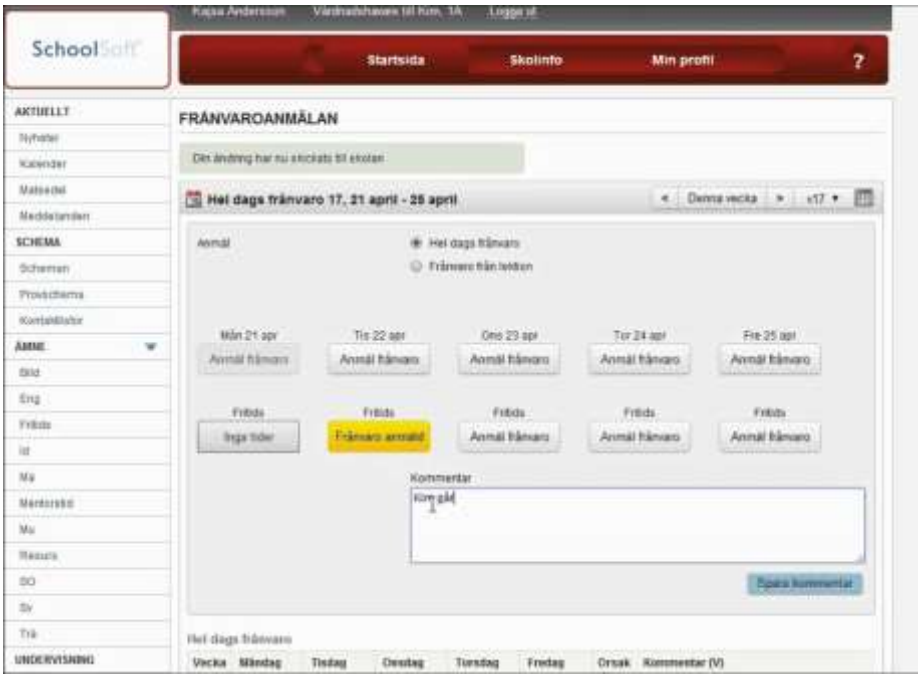


Рисунок 1.2 – Інформаційна система SchoolSoft

Moodle є однією з найбільш популярних систем управління навчальним процесом в університетах. Ця система пропонує широкий спектр функцій, включаючи можливість створення курсів, завдань, тестів та форумів для обговорення. Однією з ключових переваг Moodle є його гнучкість, яка дозволяє викладачам налаштовувати курси відповідно до їхніх потреб. [9] Крім того, система може інтегруватися з різними базами даних та іншими системами управління, що робить її ще більш універсальною. Moodle також підтримується великою спільнотою розробників та користувачів, що сприяє швидкому вирішенню проблем. Однак, система має свої недоліки. Зокрема, вона не має спеціалізованих інструментів для виховної роботи, фокусуючись переважно на академічних аспектах. Також, для повноцінного використання всіх функцій Moodle потрібно мати певний рівень технічних знань.



Рисунок 1.3 – Інформаційна система Moodle

Після детального аналізу існуючих систем управління навчальним процесом можна зробити висновок, що жодна з них не надає комплексного рішення для автоматизації виховної роботи. Хоча системи, такі як EduSys, SchoolSoft та Moodle пропонують ряд корисних функцій для академічного обліку та аналітики, їх фокус переважно зосереджений на академічних аспектах навчання. Вони не надають спеціалізованих інструментів для планування, обліку та аналізу виховних заходів, що є ключовими для інформаційної системи, яка розробляється. Також вони не можуть бути інтегровані з існуючою системою АСУ для автоматизації процесу авторизації та забезпечення єдності даних. Таким чином, існує велика потреба в розробці нової системи, яка би враховувала специфіку виховної роботи і надавала необхідні інструменти для її ефективного управління.

1.3. Інтеграція із існуючою АСУ

Програмний комплекс "Автоматизована система управління навчальним закладом" представляє собою безліч пов'язаних між собою програм, що забезпечують управління ЗВО в єдиному інформаційному просторі, і включає модулі, що працюють в середовищі Windows (навчальний модуль, деканат,

абітурієнт, методичний відділ, відділ кадрів і т.п. .д.) та WEB портал (відображення розкладу занять, успішності, навчальних планів, нарахувань сплати за гуртожиток, контроль сплати за навчання та гуртожиток, тестування студентів, запис студентів на вивчення дисциплін тощо). Вся інформація зберігається в одній спільній базі даних. [10]

Інтеграція із АСУ має на меті забезпечення безшовного потоку даних між двома системами та використання даних, збережених у АСУ, інформаційною системою для обліку студентських груп та контролю виховної роботи. Це включає дані, які пов’язані зі студентами, інформацією про групи та інші відповідні дані, які використовуються для ефективного планування та виховної роботи.

Особливість кураторської діяльності полягає в тому, що куратор одночасно є і суб'єктом діяльності, і її об'єктом. Це дозволяє йому ставити перед собою особисті цілі саморозвитку і досягати їх в процесі діяльності. Ці цілі пов'язані з формуванням педагогічних і управлінських здібностей. [11]

Оскільки куратори груп та склад груп визначаються в АСУ, а АСУ та система для проведення обліку та виховної роботи зі студентськими групами це окремі автономні інформаційні системи, то крім авторизації за допомогою АСУ ще виникає необхідність в синхронізації даних журналу куратора з АСУ.

Одним із головних викликів при інтеграції з АСУ є забезпечення узгодженості та безпеки даних. Інформаційна система повинна мати можливість безпечного доступу та отримання даних з АСУ без порушення цілісності та конфіденційності даних. Крім того, система повинна мати можливість обробляти дані у спосіб, що відповідає політиці та практиці управління даними навчального закладу.

У контексті кураторської роботи інтеграція із АСУ принесе чимало переваг. Наприклад, куратори можуть мати доступ до даних студентів у реальному часі, які можна використовувати для адаптації виховної роботи відповідно до конкретних потреб. Крім того, інтеграція посприяє

автоматизованому введенню та пошуку даних, зменшуючи ручну роботу та мінімізуючи ризик помилок.

Більше того, інтеграція гарантує, що будь-які оновлення або зміни, внесені в АСУ, такі як зміни у складі студентських груп, автоматично відображаються в інформаційній системі для обліку та контролю виховної роботи з студентськими групами. Це гарантує, що куратори завжди працюють з найактуальнішою інформацією, підвищуючи точність та ефективність своєї роботи.

Підсумовуючи, інтеграція інформаційної системи виховної роботи із АСУ не тільки покращить функціональність та надійність системи, але і гарантує, що куратори можуть виконувати свої обов'язки більш ефективно, маючи доступ до точної та своєчасної інформації. Таким чином, ця інтеграція є важливим компонентом у розробці інформаційної системи, забезпечуючи її готовність відповідати вимогам та викликам сучасної виховної роботи у вишах.

1.4. Постановка задачі

В контексті сучасного освітнього процесу, де акцент зміщується не тільки на академічну підготовку, але і на всебічний розвиток студента, виникає потреба в комплексному підході до управління виховним процесом. Аналіз предметної області та існуючих систем показав, що більшість з них не в змозі задовольнити всі потреби освітнього закладу в цьому контексті. Тому основною задачею даного проекту є розробка інформаційної системи, яка б була здатна автоматизувати та оптимізувати ряд ключових процесів.

Специфічні задачі, які необхідно вирішити, включають:

- Автоматизація процесу обліку та контролю виховної роботи студентів, що дозволить кураторам груп автоматизувати процес планування та організації виховних заходів;

- Забезпечення об'єктивності та достовірності даних шляхом впровадження механізмів перевірки введених даних та контролю їхньої точності;
- Інтеграція розробленої системи з існуючою АСУ для авторизації користувачів, що забезпечить єдність даних та зменшить ризик несанкціонованого доступу, а також допоможе синхронізувати дані про кураторів та склад груп;
- Реалізація функціоналу оцінювання особистих якостей студентів на основі даних, введених кураторами, що дозволить систематизувати інформацію та використовувати її для підвищення ефективності виховної роботи;
- Реалізація функціоналу збору та аналізу звітної інформації, який допоможе відповідальним особам з виховної роботи в оцінці ефективності цієї роботи.

Ці задачі формують фундамент для подальшого проектування та розробки системи. Вони визначають як функціональні, так і нефункціональні вимоги, і стають вихідним пунктом для оцінки ефективності розробленої системи в майбутньому.

РОЗДІЛ 2 ВИБІР ТЕХНОЛОГІЙ ТА АРХІТЕКТУРИ СИСТЕМИ

2.1. Вибір технологій розробки

Вибір мов програмування та інструментів впливає на всі аспекти системи, від швидкості розробки до її стабільності та безпеки. [12] HTML, CSS та Javascript були обрані для створення динамічного веб-інтерфейсу, що забезпечує зручність для користувача та адаптивність. Бібліотека JQuery та фреймворк Bootstrap допомагають у створенні сучасного дизайну, а PHP, як серверна мова програмування, забезпечує гнучкість та надійність у взаємодії з базою даних MySQL. PHPOffice, в свою чергу, допомагає формувати офісні документи для подальших звітів. Кожна з цих технологій була обрана на основі її специфічних переваг та здатності відповідати конкретним вимогам проекту. Розглянемо основні особливості та переваги обраних технологій.

HTML, CSS

HTML (HyperText Markup Language) є основною мовою розмітки для створення веб-сторінок. Вона формує каркас будь-якої веб-сторінки, визначаючи її структуру та зміст.

HTML використовує теги для визначення різних елементів на сторінці, таких як заголовки, параграфи, списки, посилання, зображення, таблиці та багато інших. Ці теги вказують браузеру, як відображати відповідний вміст, створюючи ієрархічну структуру документа. [13]

Сучасні версії HTML, такі як HTML5, вводять нові семантичні елементи, які допомагають розробникам створювати більш зрозумілу та доступну структуру сторінки. Це включає в себе теги, які визначають основні частини сторінки, такі як <header>, <footer>, <article> та <section>, що полегшує навігацію та доступність для користувачів з обмеженими можливостями.

HTML також надає засоби для взаємодії з користувачем через форми. Від текстових полів та кнопок до каскадних списків та перемикачів, HTML

дозволяє збирати вхідні дані від користувачів та передавати їх на сервер для обробки.

Мультимедійні можливості HTML також важливі. Вбудовані теги для аудіо та відео дозволяють інтегрувати різноманітний мультимедійний контент без потреби в сторонніх плагінах чи інструментах. [14]

Адаптивність є ще однією ключовою особливістю HTML. З допомогою різних тегів та атрибутів можна створювати дизайн, який автоматично налаштовується під різні розміри екрану та пристрої, забезпечуючи оптимальний досвід користувача.

HTML є відкритим стандартом, що підтримується всіма сучасними веб-браузерами. Це забезпечує широку сумісність та стабільність відображення вмісту на різних платформах. [15]

В контексті даного проекту, використання HTML є стратегічним вибором, що забезпечує надійну, гнучку та адаптивну структуру веб-інтерфейсу системи. Це дозволяє розробникам створювати інтуїтивно зрозумілий та зручний для користувача інтерфейс, який відповідає сучасним стандартам веб-дизайну.

```
<!DOCTYPE html>
<html lang="uk">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Назва сторінки</title>
</head>
<body>
    <!-- Тут розміщується контент сторінки -->
</body>
</html>
```

Рисунок 2.1 – Базова структура HTML-сторінки

CSS (Cascading Style Sheets) є мовою стилізації, яка використовується для опису візуального вигляду документа, написаного на мові розмітки, такій як HTML. CSS дозволяє розробникам контролювати різні аспекти відображення елементів на веб-сторінці, включаючи колір, розмір, відступи, шрифти, вирівнювання та багато іншого.

Однією з ключових особливостей CSS є його каскадна природа. Стилі можуть бути визначені на різних рівнях (глобально, на рівні елемента та інтегровано в HTML) і поєднуватися разом, дозволяючи розробникам створювати складні та гнучкі візуальні схеми. [16]

CSS також підтримує медіа-запити, які дозволяють адаптувати стилі під різні розміри екрану та типи пристроїв. Це забезпечує адаптивний дизайн, який автоматично налаштовується під різні умови відображення.

Модульність є ще однією важливою особливістю CSS. [16] Стилі можна організувати в окремі файли та повторно використовувати в різних частинах веб-сайту або навіть між різними проектами. Це сприяє ефективності розробки та компонуванню коду.

Анімації та переходи в CSS дозволяють додавати динамічні візуальні ефекти без використання Javascript. Це може включати плавні переходи по сторінках, обертання, масштабування та інші анімаційні ефекти.

```css
/* Основні стилі для body */
body {
  font-family: Arial, sans-serif;
  line-height: 1.6;
  color: #333;
}
```

Рисунок 2.2 – Приклад CSS

CSS є відкритим стандартом, який підтримується всіма сучасними веб-браузерами. Його синтаксис та властивості постійно розвиваються, додаючи нові можливості та вдосконалення для розробників.

В контексті даного проекту, використання CSS є важливим для створення зручного, адаптивного та естетично привабливого інтерфейсу. Його гнучкість, модульність та здатність до адаптації роблять його ідеальним інструментом для стилізації веб-інтерфейсу системи, забезпечуючи користувачам високоякісний досвід.

Bootstrap є одним з найпопулярніших відкритих фреймворків для розробки адаптивних веб-сайтів та веб-додатків. Він був створений та розроблений командою Twitter і вперше випущений у 2011 році. З того часу Bootstrap став стандартом в індустрії, завдяки своїм гнучким можливостям та масштабованістю. [17]

Основна мета Bootstrap полягає в тому, щоб надати розробникам швидкий старт у створенні адаптивних веб-сайтів. Він включає в себе набір готових для використання компонентів, таких як навігаційні панелі, модульні вікна, вкладки, сповіщення та багато іншого, які можна легко інтегрувати в проект.

Bootstrap використовує систему сітки, яка дозволяє розробникам легко створювати адаптивні дизайни, які автоматично підлаштовуються до різних розмірів екрану. Ця система сітки поділяє екран на 12 рівних стовпців, які можна комбінувати та масштабувати для створення унікальних дизайнів.

Фреймворк також включає в себе набір готових до використання стилів для типографії, форм, кнопок та інших елементів інтерфейсу. Це дозволяє розробникам створювати зручний та зрозумілий інтерфейс без потреби в ручному написанні великої кількості CSS-коду.

Bootstrap підтримує різні плагіни на основі jQuery, які додають додаткову функціональність, таку як каруселі, каскадні меню, перемикачі та інше. Ці плагіни можна включити або виключити за потреби, що дозволяє забезпечити оптимальну продуктивність.

Однією з ключових переваг Bootstrap є його документація та спільнота. Велика кількість посібників, прикладів та форумів забезпечують швидке вивчення та вирішення проблем, що можуть виникнути під час розробки. [18]

В контексті даного проекту, використання Bootstrap було стратегічним вибором, адже цей фреймворк дозволив швидко розробити адаптивний та зручний інтерфейс. Його готові компоненти та гнучка система сітки забезпечили ефективність розробки, дозволяючи зосередитися на унікальних функціональних вимогах системи.

Javascript та JQuery

Javascript є мовою програмування, яка в основному використовується для створення інтерактивних веб-сайтів. Вона є однією з трьох основних технологій веб-розробки поряд з HTML та CSS.

Javascript дозволяє розробникам додавати динамічний вміст на веб-сторінки, такий як анімації, взаємодія з користувачем, зміна вмісту без перезавантаження сторінки та інше. Це може включати в себе реакцію на дії користувача, такі як кліки мишею, натискання клавіш, перетягування елементів та багато іншого.

Javascript також використовується на серверній стороні через платформи, такі як Node.js. Це дозволяє створювати повноцінні веб-додатки, які працюють як на клієнтській, так і на серверній стороні, використовуючи одну мову програмування. [19]

Однією з ключових особливостей Javascript є його асинхронна природа. За допомогою конструкцій, таких як проміси та асинхронні функції, розробники можуть виконувати операції, які можуть займати певний час, такі як завантаження даних з сервера, без блокування інших частин сторінки.

Javascript також підтримує об'єктно-орієнтоване програмування, що дозволяє створювати модульний та легко супроводжуваний код. Це включає в себе використання класів, спадкування, інкапсуляції та інших парадигм, які сприяють створенню масштабованих та гнучких додатків. [20]

В контексті даного проекту, Javascript був обраний для створення динамічного та інтерактивного інтерфейсу. Його гнучкість та широкі можливості зробили його ідеальним вибором для реалізації складних функціональних вимог проекту.

jQuery є швидкою, невеликою та багатофункціональною бібліотекою JavaScript, яка спрощує такі речі, як обробка подій, анімація та взаємодія з AJAX. Вона створена з метою полегшення клієнтської частини веб-розробки.

Однією з основних переваг бібліотеки jQuery є її простота використання. Вона надає зручний синтаксис, який дозволяє розробникам легко знаходити та маніпулювати HTML-елементами, використовуючи CSS-селектори. [21] Це може суттєво прискорити процес розробки, зменшуючи кількість коду, який потрібно написати.

jQuery також відома своєю кросбраузерною сумісністю. Вона надає абстракцію від різниці між різними веб-браузерами, дозволяючи розробникам писати код, який працює чітко в усіх основних браузерах без потреби в специфічних обходах.

Бібліотека також включає в себе потужні засоби для анімації. З її допомогою можна легко створювати плавні переходи, слайд-шоу, згортання чи розгортання елементів та інші візуальні ефекти. Це дозволяє створювати динамічні та привабливі інтерфейси без великих зусиль.

jQuery також підтримує AJAX, технологію, яка дозволяє веб-додаткам взаємодіяти з сервером без перезавантаження сторінки. [22] Це використовується для завантаження даних, відправки форм, оновлення вмісту в реальному часі та інших інтерактивних функцій.

В контексті даного проекту, використання jQuery дозволило швидко та ефективно створити інтерактивний інтерфейс, який відповідає сучасним стандартам веб-дизайну. Його гнучкість, легкість використання та широкі можливості зробили його важливим інструментом у розробці даної системи.

PHP та PHPOffice

PHP (Hypertext Preprocessor) є серверною мовою програмування, яка широко використовується для розробки веб-додатків. [23] Вона є однією з найпопулярніших мов для веб-розробки і використовується в мільйонах веб-сайтів по всьому світу.

Однією з ключових особливостей PHP є її здатність взаємодіяти з різними системами управління базами даних, такими як MySQL. Це дозволяє розробникам легко створювати динамічні веб-сайти, які можуть зберігати, отримувати та маніпулювати даними в реальному часі.

PHP є мовою скриптів на стороні сервера, що означає, що весь код виконується на сервері, а не в браузері користувача. [24] Це забезпечує додатковий рівень безпеки, оскільки код не відображається клієнтам і не може бути змінений ними.

Мова також відома своєю гнучкістю та широкими можливостями. Вона підтримує об'єктно-орієнтоване, процедурне та функціональне програмування, дозволяючи розробникам вибирати підхід, який найкраще підходить для конкретного проекту. [25]

PHP має велику спільноту розробників та багато відкритих ресурсів, включаючи бібліотеки, фреймворки, інструменти розробки та документацію. Це сприяє швидкому вивченню мови та її ефективному використанню в проектах різного рівня складності.

В контексті даного проекту, використання PHP було стратегічним вибором для реалізації серверної частини системи. Його можливості з взаємодією з базами даних, гнучкість та надійність зробили його ідеальним інструментом для створення складної та масштабованої веб-платформи.

PHPOffice [26] є набором бібліотек, які дозволяють розробникам читати, писати, змінювати та створювати документи в різних форматах офісних файлів, таких як Word, Excel, PowerPoint, безпосередньо з PHP. Це відкритий проект, який активно підтримується спільнотою.

Ось ключові особливості PHPOffice, які зробили його важливим інструментом у даному проекті:

- Сумісність з форматами: PHPOffice підтримує різні формати офісних файлів, включаючи найпопулярніші формати Word та Excel. Це дозволяє генерувати документи, які можна відкрити та редагувати в більшості офісних пакетів.
- Гнучкість: бібліотека надає широкий спектр функцій для маніпулювання з даними, форматування тексту, роботи з стилями, вставки зображень та іншого. Це дозволяє створювати складні та налаштовані документи з PHP.
- Автоматизація: з допомогою PHPOffice можна автоматизувати процеси створення та обробки документів, що забезпечує ефективність та зменшує можливість помилок. [27]
- Інтеграція з веб-додатками: PHPOffice легко інтегрується з веб-додатками на PHP, дозволяючи генерувати, зберігати та передавати документи безпосередньо через веб-інтерфейс.

Використання PHPOffice в даному проекті дозволило створити потужний та гнучкий інструмент для генерації звітів, який відповідає специфічним вимогам системи. Його можливості з автоматизації та налаштування забезпечили ефективний процес формування таблиць у Word та Excel-файлах, що є однією із частин функціональності системи.

MySQL та phpMyAdmin

MySQL є однією з найпопулярніших відкритих систем управління реляційними базами даних (RDBMS). Вона використовується в якості основного засобу зберігання даних у багатьох інформаційних веб-системах.

Ця система відзначається високою продуктивністю, надійністю та гнучкістю. [28] Вона підтримує стандарт SQL, що дозволяє легко створювати, оновлювати та надсилати запити до бази даних. Завдяки широкому спектру інструментів оптимізації та налаштування, MySQL може бути налаштована для високої продуктивності в різних сценаріях використання.

Вибір MySQL для даного проекту був обґрунтований її стабільністю, швидкістю та легкістю інтеграції з різними мовами програмування, такими як PHP. Ця сумісність забезпечує гладку взаємодію між серверною частиною додатку та базою даних.

Крім того, MySQL має велику спільноту розробників та багато доступних ресурсів, що полегшує розробку та підтримку. [29] Її відкритий характер також означає, що вона є безкоштовною для використання в більшості сценаріїв, що робить її доступною та економічно ефективною опцією.

Всі ці фактори разом зробили MySQL ідеальним вибором для зберігання та управління даними в даному проекті, забезпечуючи надійну, швидку та гнучку платформу, яка відповідає потребам системи.

phpMyAdmin є вільним інструментом для адміністрування MySQL через веб-інтерфейс. Цей інструмент є незамінним для розробників та адміністраторів баз даних, які шукають зручний спосіб керування своїми базами даних без необхідності прямої взаємодії з командним рядком MySQL.

phpMyAdmin надає користувачам можливість переглядати, створювати, редагувати та видаляти таблиці, бази даних, індекси, користувачів та інші об'єкти бази даних за допомогою графічного інтерфейсу. [30] Це значно спрощує процес розробки та підтримки, особливо для тих, хто може бути менш знайомий з синтаксисом SQL.

Вибір phpMyAdmin у даному проекті був обґрунтований його широкою підтримкою, надійністю та легкістю використання. Його інтеграція з PHP та MySQL забезпечує гладку взаємодію з іншими технологіями, використаними в проекті, і дозволяє швидко вносити зміни в структуру бази даних без необхідності ручного написання SQL-запитів.

phpMyAdmin також має ряд додаткових функцій, таких як імпорт та експорт даних, виконання SQL-запитів, управління користувачами та привілеями, а також можливість створювати та використовувати збережені процедури та тригери. Ці функції роблять його потужним інструментом для розробки та адміністрування баз даних. [31]

Загалом, використання phpMyAdmin в даному проекті забезпечило зручний та ефективний спосіб управління базою даних, що сприяло швидкій розробці та впровадженню системи.

Вибір середовища програмування

При розробці сучасних веб-додатків вибір відповідного середовища програмування є критичним аспектом, який може вплинути на продуктивність, гнучкість та підтримку необхідних технологій. У контексті даного проекту було обрано Microsoft Visual Studio Code (VS Code) [32] як основне середовище розробки, і цей вибір був зроблений з урахуванням декількох ключових факторів.

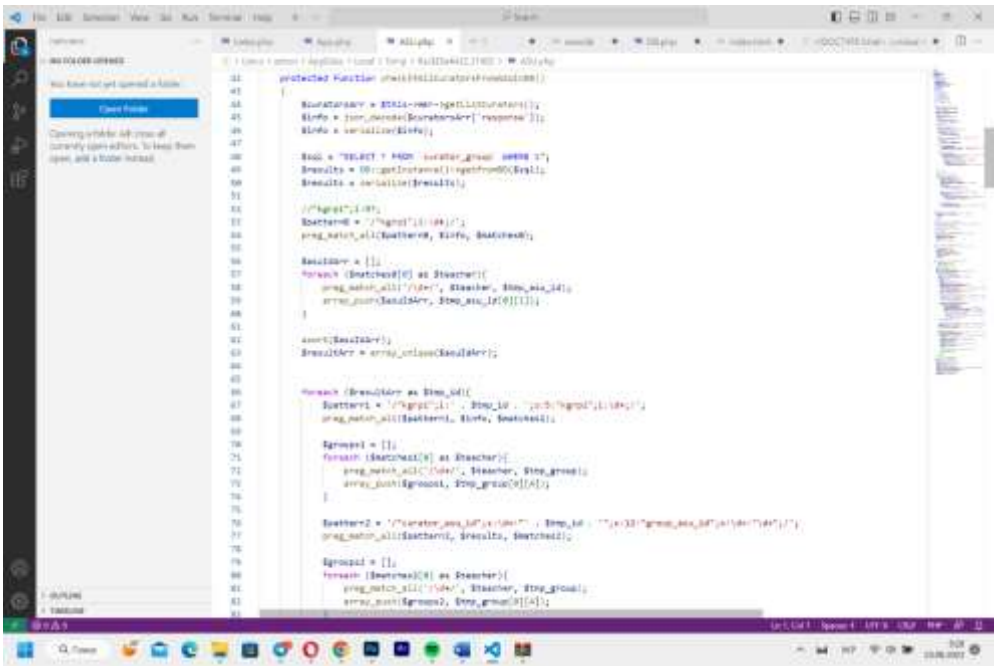


Рисунок 2.3 – Зовнішній вигляд середовища розробки VS Code

VS Code є безкоштовним редактором коду від Microsoft, який поєднує в собі легкість текстового редактора з потужними можливостями інтегрованого середовища розробки (IDE). [33] Його популярність серед розробників по всьому світу зростає завдяки комбінації гнучкості, продуктивності та підтримки сучасних технологій.

Однією з найбільших переваг VS Code є його підтримка великої кількості мов програмування, включаючи HTML, CSS, JavaScript, PHP, що були використані в проекті. Ця підтримка забезпечує плавний перехід між різними частинами проекту та сприяє швидкій розробці.

Через відкритий магазин розширень, розробники можуть легко додавати додаткові функції та інструменти, що специфічні для їх потреб. Це дозволяє гнучко налаштовувати редактор під свої особисті потреби та стиль роботи, від вбудованої підтримки системи контролю версій Git до інтеграції з різними засобами збірки та тестування.

VS Code також відзначається своєю кросплатформеністю, будучи доступним для Windows, macOS та Linux. Це забезпечує чіткий процес розробки на різних операційних системах, дозволяючи командам співпрацювати ефективно, незалежно від їхнього обладнання. [34]

Легкість та швидкість редактора сприяють високій продуктивності розробки, дозволяючи зосередитися на написанні коду без зайвих відволікань. Його інтуїтивний інтерфейс, гнучкі налаштування та вбудовані інструменти для налагодження коду роблять його вибором багатьох професійних розробників.

Вибір Microsoft Visual Studio Code як основного середовища програмування для даного проекту був обґрунтований його гнучкістю, підтримкою необхідних технологій, можливістю налаштування та інтеграції з іншими інструментами. Це забезпечило ефективний процес розробки та сприяло успішній реалізації проекту, відповідаючи високим стандартам якості та продуктивності, які вимагаються в сучасному світі веб-розробки.

2.2. Архітектура системи

Вибір архітектури для розробки інформаційної системи є одним з найважливіших аспектів при розробці – від швидкості розробки системи до безпеки даних. У контексті даного проекту, клієнт-серверна архітектура була обрана як найбільш оптимальний варіант. Ця модель архітектури відома своєю

надійністю, масштабованістю та гнучкістю, що є критично важливим для системи, що зберігає та обробляє великі об'єми даних.

Клієнт-серверна архітектура [35] дозволяє розділити функціональність між серверною та клієнтською частинами. Серверна частина відповідає за обробку даних та взаємодію з базою даних. Це забезпечує централізоване управління, що є важливим для забезпечення цілісності даних.

Використання серверної частини для обробки даних про облік груп також забезпечує високий рівень безпеки. Всі запити проходять через сервер, де вони можуть бути перевірені на несанкціонований доступ до даних.

Щодо інтеграції з автоматизованою системою управління університетом (АСУ), це дозволяє створити єдину точку входу для користувачів, забезпечуючи їх ідентифікацію та авторизацію. Це не тільки забезпечує високий рівень безпеки, але і значно полегшує процес взаємодії користувача з системою, оскільки не потрібно пам'ятати декілька даних від облікових записів.

Крім того, клієнт-серверна архітектура дозволяє легко додавати нові функціональні модулі або змінювати існуючі без втрати цілісності системи. [36] Це робить архітектуру найбільш відповідною для інформаційних систем, які можуть розширюватися або модифікуватися в майбутньому.

Таким чином, вибір клієнт-серверної архітектури для даної інформаційної системи був зроблений на основі ряду ключових факторів, таких як потреба в централізованому управлінні, висока безпека даних, можливість масштабування та потреба в гнучкості для подальшого розвитку системи.

Розробка інформаційної системи для обліку та котролю виховної роботи з студентськими групами вимагає чіткого підходу до вибору технологій. Саме тому була розроблена концептуальна схема даної інформаційної системи (рис 2.4).

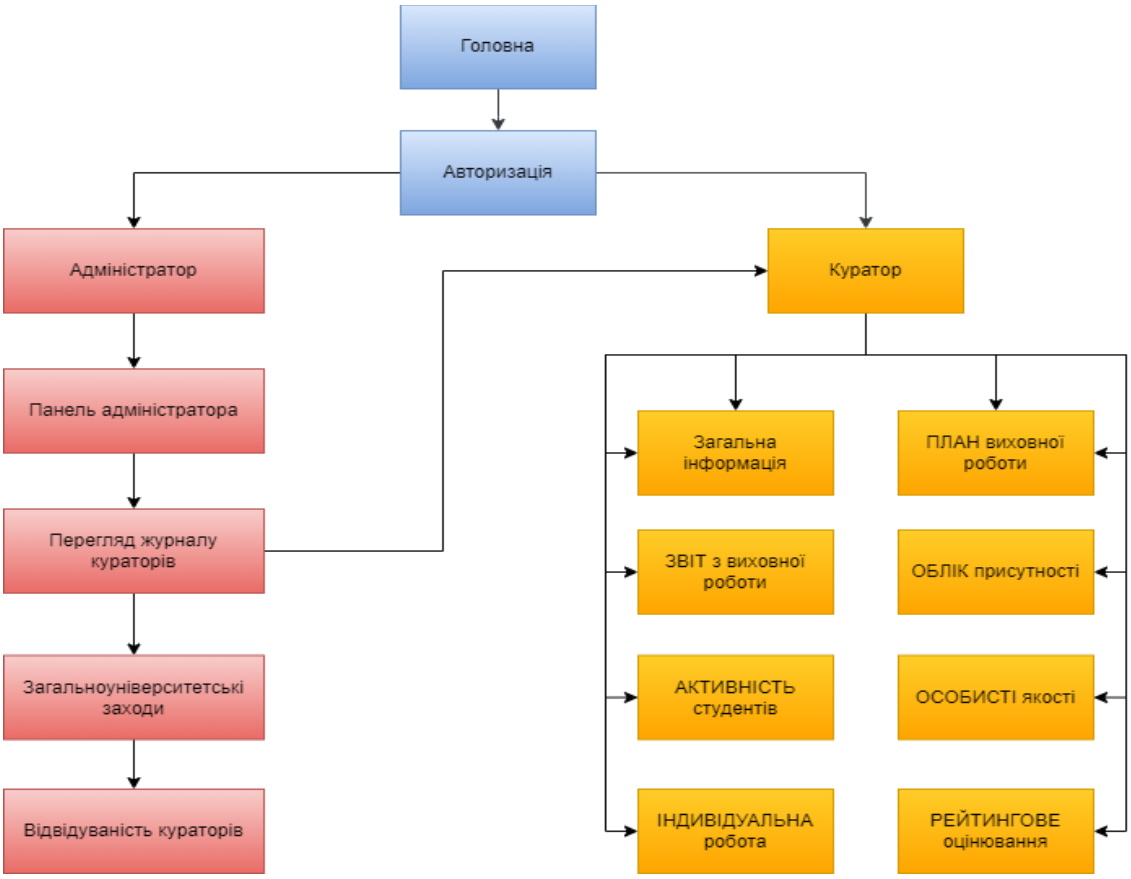


Рисунок 2.4 – Загальна структура інформаційної системи

У таблиці 2.1 зроблено опис акторів, які є зовнішніми сутностями стосовно системи та взаємодіють з проектованою системою для розв'язання певних задач.

Таблиця 2.1 – Актори ІС електронного навчання

Актор	**Короткий опис**
Адміністратор	Адміністратор забезпечує роботу інформаційної системи та надає допомогу користувачам у вирішенні проблем із системою, якщо такі виникають.
Куратор	Користувач, що веде облік студентських груп, присутності студентів, участі у заходах та проводить індивідуальну роботу.

Система повинна надавати можливість для обліку студентів, їхньої присутності, участі в заходах та індивідуальної роботи. Це дозволить кураторам систематизувати дані та відстежувати динаміку розвитку студентів.

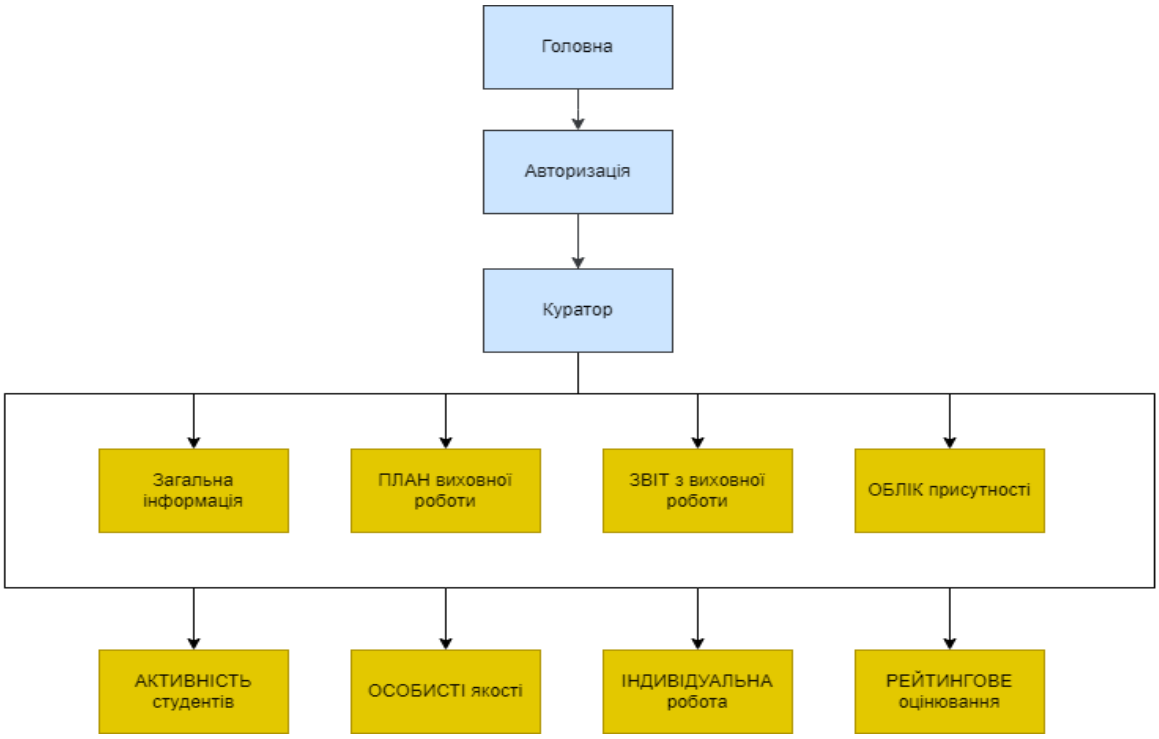


Рисунок 2.4 – Структура інформаційної системи для користувача з роллю куратора

Адміністратор системи надає користувачам технічну підтримку та слідкує за справністю функціонування системи. Він також має права на перегляд журналів, але не має дозволу на їх редагування (рис. 2.5).



Рисунок 2.5 – Структура інформаційної системи для користувача з роллю адміністратора

Користувач взаємодіє з інформаційною системою, використовуючи веб-сторінки сайту для заповнення електронного журналу куратора. Користувач надсилає запити та отримує реакції від веб-сервера. Сервер забезпечує користувацький інтерфейс, а також використовує інтерфейс бази даних для доступу, модифікації, видалення та зміни даних. (рис. 2.5).

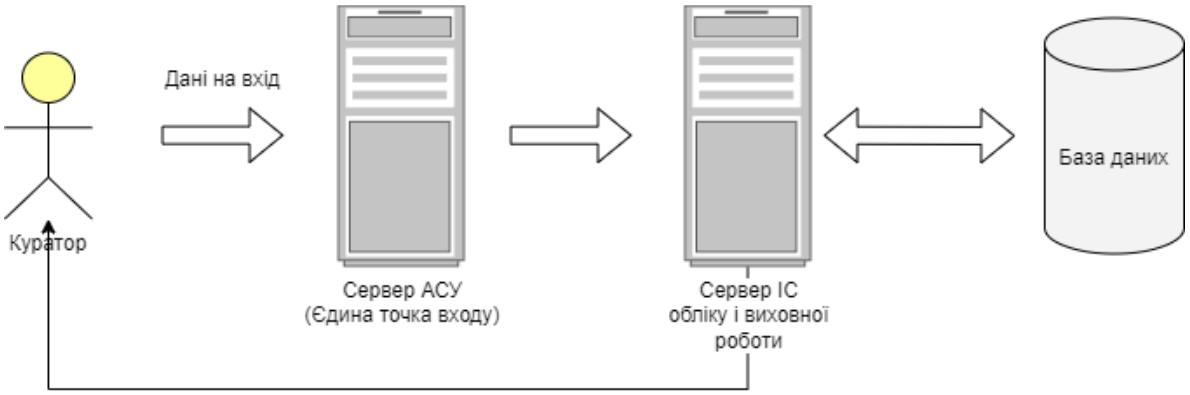


Рисунок 2.5 – Архітектура інформаційної системи

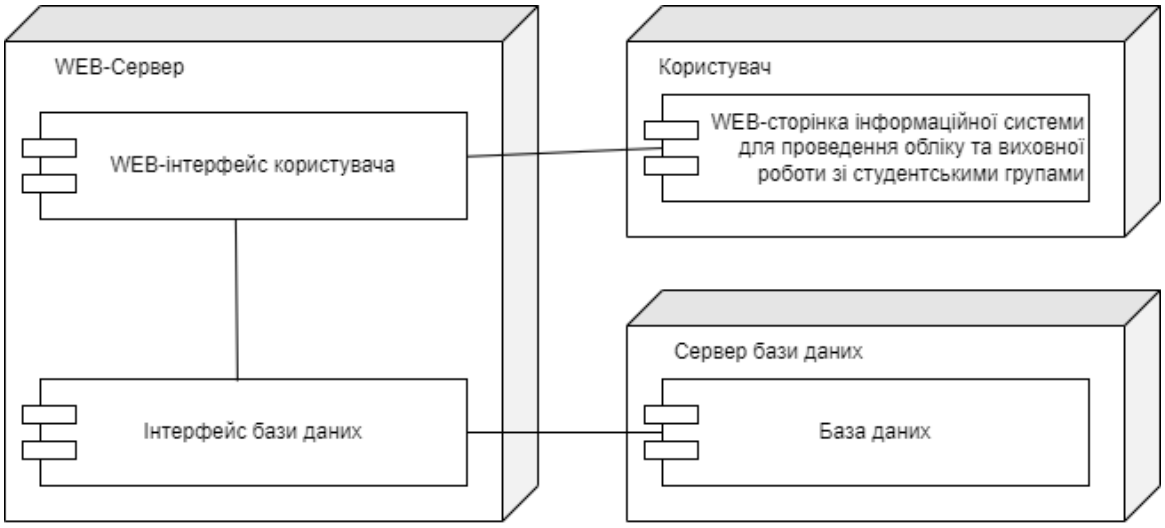


Рисунок 2.6 – Діаграма розгортання інформаційної системи

2.3. Проектування бази даних

Проектування бази даних є одним з найважливіших етапів розробки інформаційної системи для обліку та контролю виховної роботи з студентськими групами. Від ефективності структури бази даних залежить швидкість доступу до інформації, можливість масштабування та надійність системи в цілому. [37]

База даних розроблена на платформі MySQL і має реляційну структуру. Вона складається з декількох таблиць, кожна з яких відповідає за певний аспект виховної роботи. Наприклад, є таблиці для зберігання інформації про студентів, їхніх батьків, планованих та проведених заходах, а також рейтингових оцінках.

Для забезпечення цілісності даних та зручності їх обробки, використовуються різні типи зв'язків між таблицями, включаючи "один до одного", "один до багатьох" та "багато до багатьох". [38] Це дозволяє ефективно вести облік та аналізувати виховну роботу на різних рівнях - від індивідуальної роботи зі студентами до аналізу ефективності виховної роботи на рівні групи чи навіть університету.

Для адміністрування бази даних використовується система phpMyAdmin, яка дозволяє легко управляти структурою бази, проводити резервне копіювання та оптимізацію даних.

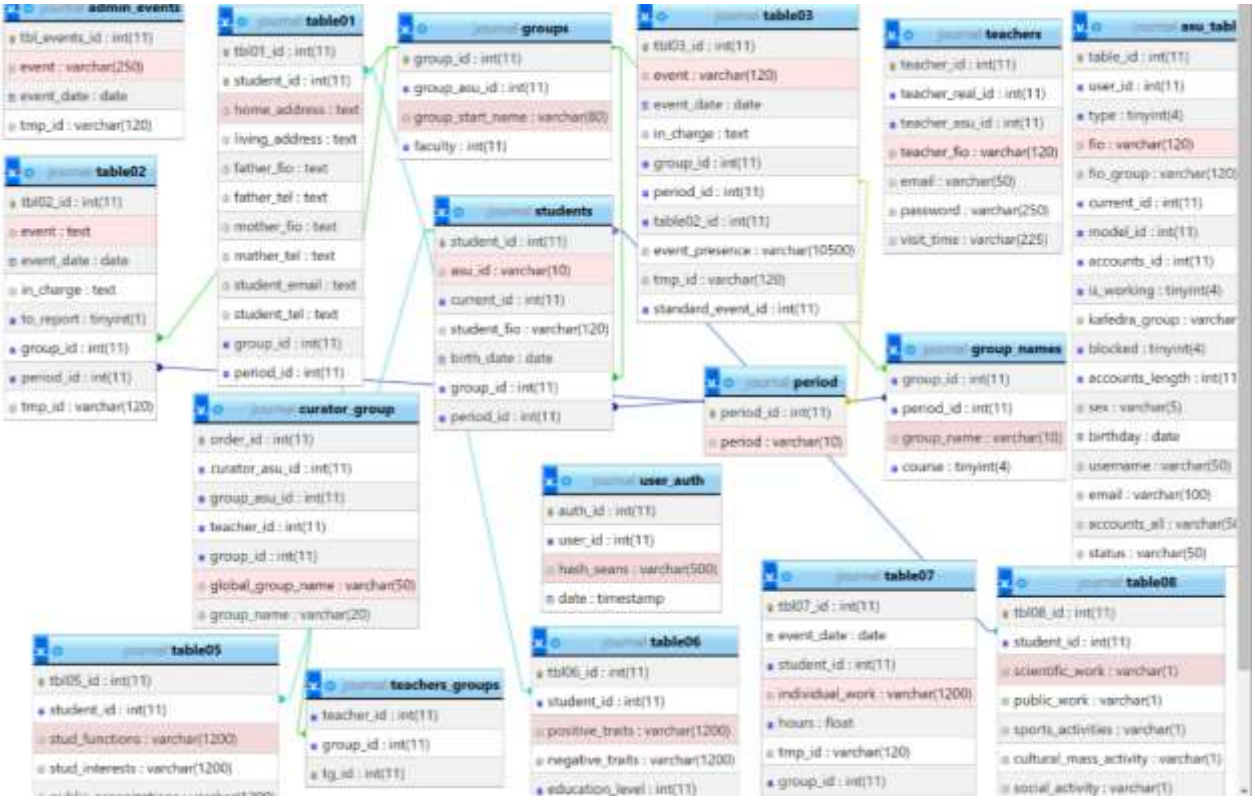


Рисунок 2.8 – Схема даних ІС

Основними сутностями інформаційної системи виділено наступні:

- "Адміністратор", що зберігає відомості про адміністратора ресурсу;
- "Загальна інформація", що зберігає інформацію про персональні дані студентів та відомості про батьків;
- "План виховної роботи", що зберігає інформацію про заплановані куратором заходи;

- "Звіт з виховної роботи", що зберігає інформацію про проведені куратором заходи;
- "Облік присутності", що зберігає інформацію про присутність студентів на заходах;
- "Активність студентів", що зберігає інформацію про громадську активність студентів;
- "Особисті якості", зберігає інформацію про особисті якості студентів;
- "Індивідуальна робота", що зберігає інформацію про проведену куратором індивідуальну роботу зі студентами групи;
- "Рейтингове оцінювання", зберігає інформацію про рейтингове оцінювання студентів за 5-бальною шкалою.

2.4. Забезпечення об'єктивності та достовірності даних

Високонавантажені системи – це системи, які зазвичай оптимізовані для обробки великої кількості запитів або транзакцій в реальному часі. Ці системи мають велику кількість користувачів та/або великий обсяг даних для обробки, і вимагають відповідних архітектурних рішень та технологій для забезпечення швидкої та надійної роботи. [41] Для досягнення високої продуктивності та масштабованості високонавантажені системи часто використовують розподілені обчислення та кластеризацію серверів, що дозволяє розподілити навантаження між багатьма серверами та забезпечити відмовостійкість. Також високонавантажені системи можуть використовувати кешування даних, оптимізацію запитів та інші техніки для підвищення продуктивності та зниження навантаження на сервери.

Інформаційна система розроблена таким чином, щоб ефективно обробляти високі навантаження, забезпечуючи при цьому об'єктивність і надійність даних за допомогою надійних механізмів верифікації та контролю. Однією із ключових особливостей системи є процес перевірки даних на стороні

клієнта, який відіграє важливу роль у підтримці цілісності даних і запобіганні зламуванню системи за допомогою уразливих SQL-ін'єкцій. [39]

Наприклад, коли куратор планує захід на 2 вересня 2024 року, але обирає навчальний період 2023-2024 рр., клієнтська перевірка системи відобразить помилку, яка вкаже на те, що дата не відповідає обраному навчальному періоду. Ця попередня перевірка є захистом від помилок при введенні даних і гарантує, що вся інформація відповідає академічному календареві.

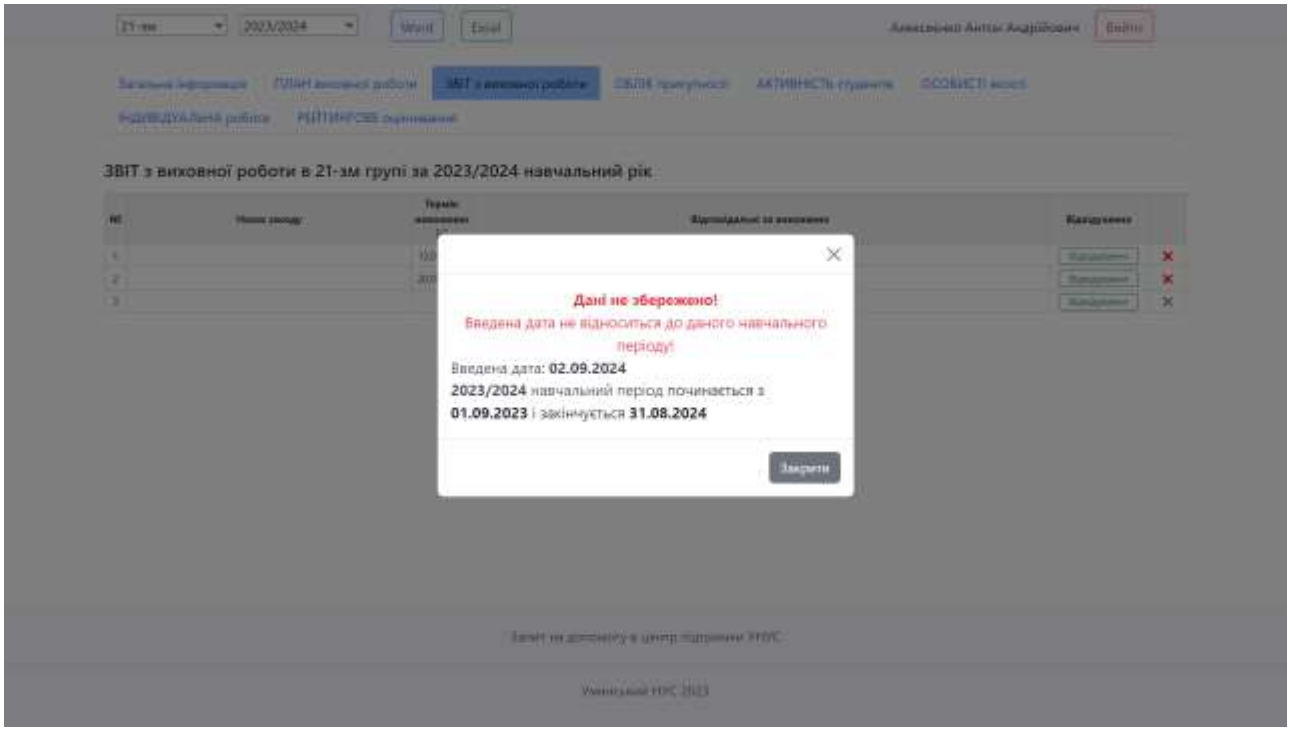


Рисунок 2.9 – Повідомлення про помилку дати, що не стосується до даного навчального періоду

Перевірка на стороні клієнта поширюється на різні функціональні сторінки системи, включаючи "План виховної роботи" та "Звіт з виховної роботи". Перевіряючи дані перед відправкою на сервер, система знижує ризик помилок і несанкціонованих маніпуляцій з даними. Такий підхід не тільки покращує досвід користувача, забезпечуючи миттєвий зворотній зв'язок, але й зменшує навантаження на сервер, оскільки дані обробляються на стороні клієнта, а далі у базу даних потрапляють тільки перевірені дані.

Крім того, для запобігання SQL-ін'єкціям та іншим формам маніпуляцій з даними в системі використовуються підготовлені запити та методи обробки вхідних даних. Ці заходи є частиною комплексної стратегії безпеки, яка захищає цілісність даних системи та запобігає зловмисному використанню. [40]

Завдяки цим механізмам система забезпечує високий рівень точності та безпеки даних навіть в умовах високого навантаження, що є важливим для надійної роботи системи в цілому.

РОЗДІЛ 3 РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ ОБЛІКУ ТА КОНТРОЛЮ ВИХОВНОЇ РОБОТИ З СТУДЕНТСЬКИМИ ГРУПАМИ

3.1. Розробка бази даних

Розроблена база даних містить наступні сутності (рисунок):

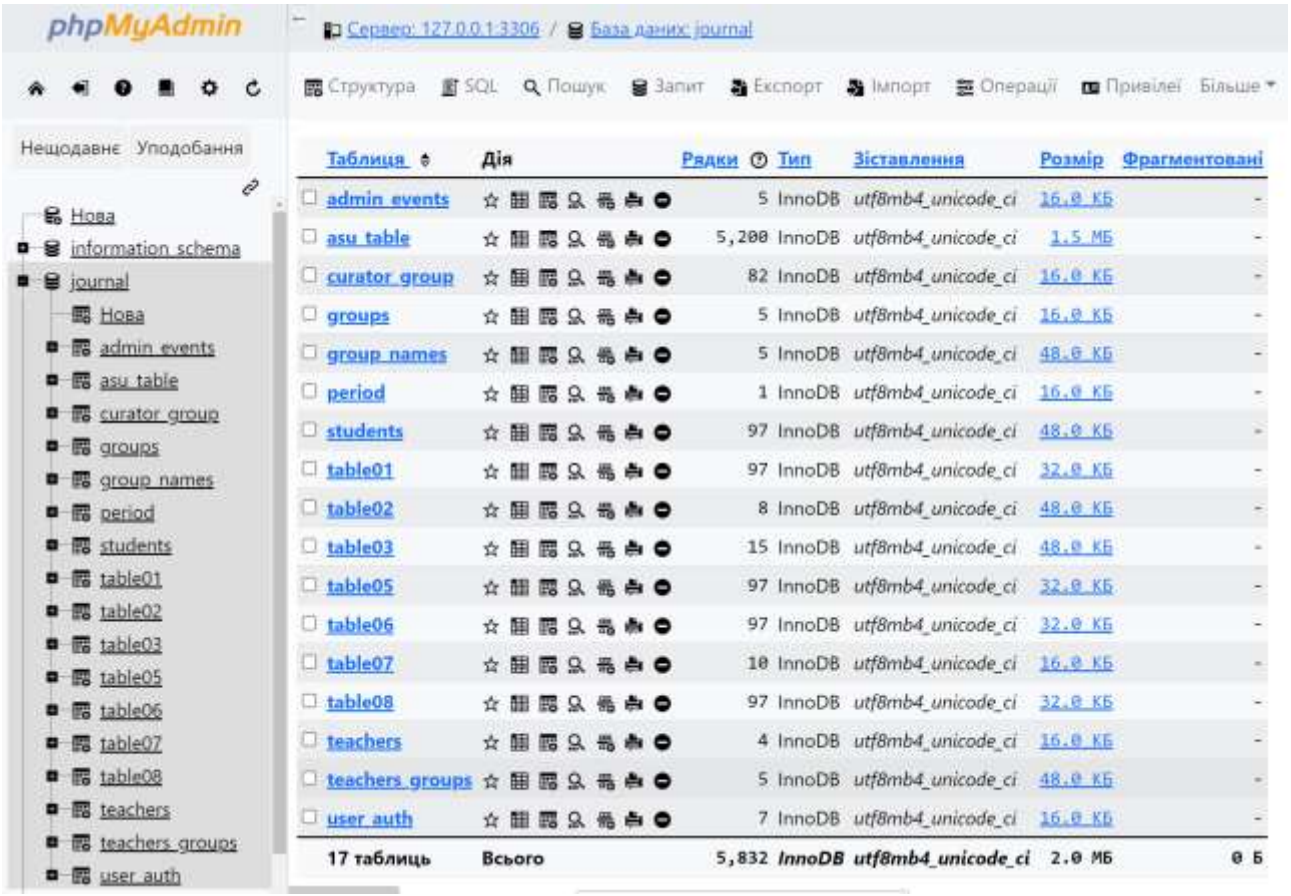


Рисунок 3.1 – Архітектура БД інформаційної системи

Сутність "admin_events" має атрибути:

- tbl_events_id – ідентифікатор події в системі, має тип int(11), не може бути NULL;
- event – назва події, має тип varchar(250), може бути NULL;
- event_date – дата події, має тип date, не може бути NULL;
- tmp_id – тимчасовий ідентифікатор, має тип varchar(120), може бути NULL.

Сутність “asu_table” має атрибути:

- table_id – ідентифікатор таблиці, має тип int(11), не може бути NULL;

- user_id – ідентифікатор користувача, має тип int(11), може бути NULL;
- type – тип, має тип tinyint(4), не може бути NULL;
- fio – повне ім'я, має тип varchar(120), може бути NULL;
- fio_group – група повного імені, має тип varchar(120), може бути NULL;
- current_id – поточний ідентифікатор, має тип int(11), може бути NULL;
- model_id – ідентифікатор моделі, має тип int(11), може бути NULL;
- accounts_id – ідентифікатор облікового запису, має тип int(11), може бути NULL;
- is_working – статус роботи, має тип tinyint(4), не може бути NULL;
- kafedra_group – група кафедри, має тип varchar(120), може бути NULL;
- blocked – статус блокування, має тип tinyint(4), не може бути NULL;
- accounts_length – довжина облікових записів, має тип int(11), не може бути NULL;
- sex – стать, має тип varchar(5), може бути NULL;
- birthday – дата народження, має тип date, не може бути NULL;
- username – ім'я користувача, має тип varchar(50), не може бути NULL;
- email – електронна адреса, має тип varchar(100), не може бути NULL;
- accounts_all – всі облікові записи, має тип varchar(500), може бути NULL;
- status – статус, має тип varchar(50), може бути NULL.

Сутність "curator_group" має атрибути:

- order_id – порядковий ідентифікатор, має тип int(11), не може бути NULL;
- curator_asu_id – ідентифікатор куратора в ASU, має тип int(11), не може бути NULL;
- group_asu_id – ідентифікатор групи в ASU, має тип int(11), не може бути NULL;
- teacher_id – ідентифікатор вчителя, має тип int(11), може бути NULL;
- group_id – ідентифікатор групи, має тип int(11), може бути NULL;
- global_group_name – глобальна назва групи, має тип varchar(50), може бути NULL;

- group_name – назва групи, має тип varchar(20), може бути NULL.

Сутність "groups" має атрибути:

- group_id – ідентифікатор групи, має тип int(11), не може бути NULL;
- group_asu_id – ідентифікатор групи в ASU, має тип int(11), не може бути NULL;
- group_start_name – початкова назва групи, має тип varchar(80), не може бути NULL;
- faculty – ідентифікатор факультету, має тип int(11), може бути NULL.

Сутність "groups_names" має атрибути:

- group_id – ідентифікатор групи, має тип int(11), не може бути NULL;
- period_id – ідентифікатор періоду, має тип int(11), не може бути NULL;
- group_name – назва групи, має тип varchar(10), не може бути NULL;
- course – номер курсу, має тип tinyint(4), не може бути NULL.

Сутність "period" має атрибути:

- period_id – ідентифікатор періоду, має тип int(11), не може бути NULL;
- period – назва періоду, має тип varchar(10), не може бути NULL.

Сутність "table01" має наступні атрибути:

- tbl01_id – ідентифікатор запису, має тип int(11), є первинним ключем, не може бути NULL;
- student_id – ідентифікатор студента, має тип int(11), не може бути NULL;
- home_address – домашня адреса студента, має тип text, може бути NULL;
- living_address – адреса проживання студента, має тип text, може бути NULL;
- father_fio – повне ім'я батька студента, має тип text, може бути NULL;
- father_tel – телефон батька, має тип text, може бути NULL;
- mother_fio – повне ім'я матері студента, має тип text, може бути NULL;
- mather_tel – телефон матері, має тип text, може бути NULL;
- student_email – електронна пошта студента, має тип text, може бути NULL;
- student_tel – телефон студента, має тип text, може бути NULL;

- group_id – ідентифікатор групи, має тип int(11), може бути NULL;
- period_id – ідентифікатор періоду, має тип int(11), може бути NULL.

Сутність "table02" має наступні атрибути:

- tbl02_id – ідентифікатор запису, має тип int(11), є первинним ключем, не може бути NULL;
- event – опис події, має тип text, не може бути NULL;
- event_date – дата події, має тип date, не може бути NULL;
- in_charge – особа, відповідальна за подію, має тип text, не може бути NULL;
- to_report – флаг, який вказує, чи потрібно звітувати про подію. Має тип tinyint(1), не може бути NULL;
- group_id – ідентифікатор групи, до якої відноситься подія, має тип int(11), не може бути NULL;
- period_id – ідентифікатор періоду, до якого відноситься подія, має тип int(11), не може бути NULL;
- tmp_id – тимчасовий ідентифікатор, має тип varchar(120), може бути NULL.

Сутність "table03" має наступні атрибути:

- tbl03_id – ідентифікатор запису, має тип int(11), є первинним ключем, не може бути NULL;
- event – назва події, має тип varchar(120), не може бути NULL;
- event_date – дата проведення події, має тип date, не може бути NULL, за замовчуванням установлено "2000-01-01";
- in_charge – особа, відповідальна за подію, має тип text, не може бути NULL;
- group_id – ідентифікатор групи, до якої належить подія, має тип int(11), не може бути NULL;
- period_id – ідентифікатор періоду, до якого належить подія, має тип int(11), не може бути NULL;

- table02_id – ідентифікатор, який зв'язує з записом в таблиці "table02", має тип int(11), не може бути NULL;
- event_presence – інформація про присутність на події, має тип varchar(10500), може бути NULL;
- tmp_id – тимчасовий ідентифікатор, має тип varchar(120), може бути NULL;
- standard_event_id – стандартний ідентифікатор події, має тип int(11), може бути NULL.

Сутність "table05" має наступні атрибути:

- tbl05_id – ідентифікатор запису, має тип int(11), є первинним ключем, не може бути NULL. Це поле автоматично інкрементується;
- student_id – ідентифікатор студента, має тип int(11), не може бути NULL;
- stud_functions – функціональні обов'язки студента або діяльність, в якій він бере участь, має тип varchar(1200), може бути NULL;
- stud_interests – інтереси студента, має тип varchar(1200), може бути NULL;
- public_organizations – публічні організації, в яких студент є членом, має тип varchar(1200), може бути NULL.

Сутність "table06" має наступні атрибути:

- tbl06_id – ідентифікатор запису, має тип int(11), є первинним ключем, не може бути NULL. Це поле автоматично інкрементується;
- student_id – ідентифікатор студента, має тип int(11), не може бути NULL;
- positive_traits – позитивні риси студента, має тип varchar(1200), може бути NULL;
- negative_traits – негативні риси студента, має тип varchar(1200), може бути NULL;
- education_level – рівень освіти студента, має тип int(11), не може бути NULL. По замовчуванню має значення "5".

Сутність "table07" має наступні атрибути:

- tbl07_id – ідентифікатор запису, має тип int(11), є первинним ключем, не може бути NULL. Це поле автоматично інкрементується;
- event_date – дата події, має тип date, не може бути NULL. По замовчуванню має значення "2000-01-01";
- student_id – ідентифікатор студента, має тип int(11), може бути NULL;
- individual_work – опис індивідуальної роботи або завдань, має тип varchar(1200), може бути NULL;
- hours – кількість годин, витрачених на індивідуальну роботу, має тип float, може бути NULL;
- tmp_id – тимчасовий ідентифікатор, може використовуватися для тимчасового зберігання даних, має тип varchar(120), може бути NULL;
- group_id – ідентифікатор групи студента, має тип int(11), може бути NULL;
- period_id – ідентифікатор періоду часу (наприклад, семестру або року), має тип int(11), може бути NULL.

Сутність "table08" має наступні атрибути:

- tbl08_id – ідентифікатор запису, має тип int(11), є первинним ключем, не може бути NULL. Це поле автоматично інкрементується;
- student_id – ідентифікатор студента, має тип int(11), може бути NULL;
- scientific_work – оцінка наукової роботи студента, має тип varchar(1), не може бути NULL. По замовчуванню має значення "5";
- public_work – оцінка громадської роботи студента, має тип varchar(1), не може бути NULL. По замовчуванню має значення "5";
- sports_activities – оцінка спортивної активності студента, має тип varchar(1), не може бути NULL. По замовчуванню має значення "5";
- cultural_mass_activity – оцінка участі в культурно-масових заходах, має тип varchar(1), не може бути NULL. По замовчуванню має значення "5";
- social_activity – оцінка соціальної активності студента, має тип varchar(1), не може бути NULL. По замовчуванню має значення "5";

- total_results – загальний результат оцінювання активності студента, має тип varchar(1), не може бути NULL. По замовчуванню має значення "5";
- group_id – ідентифікатор групи студента, має тип int(11), може бути NULL;
- period_id – ідентифікатор періоду часу (наприклад, семестру або року), має тип int(11), може бути NULL.

Сутність "teachers" має наступні атрибути:

- teacher_id – ідентифікатор викладача, має тип int(11), є первинним ключем, не може бути NULL. Це поле автоматично інкрементується;
- teacher_real_id – реальний ідентифікатор викладача (можливо, в іншій системі), має тип int(11), може бути NULL;
- teacher_asu_id – ідентифікатор викладача в системі ASU (Автоматизована Система Управління), має тип int(11), може бути NULL;
- teacher_fio – ПІБ викладача, має тип varchar(120), може бути NULL;
- email – електронна пошта викладача, має тип varchar(50), не може бути NULL;
- password – пароль для входу в систему, має тип varchar(250), не може бути NULL;
- visit_time – час останнього візиту викладача в систему, має тип varchar(225), може бути NULL.

Сутність "teachers_groups" має наступні атрибути:

- tg_id – ідентифікатор запису, має тип int(11), є первинним ключем, не може бути NULL. Це поле автоматично інкрементується.
- teacher_id – ідентифікатор вчителя, має тип int(11), не може бути NULL. Це поле також є індексом, який забезпечує швидший доступ до даних.
- group_id – ідентифікатор групи, має тип int(11), не може бути NULL. Це поле також є індексом.

Сутність "user_auth" має наступні атрибути:

- auth_id – ідентифікатор запису, має тип int(11), є первинним ключем, не може бути NULL.

- user_id – ідентифікатор користувача, має тип int(11), не може бути NULL. Це використовується для зв'язування з іншими сутностями, що містять інформацію про користувача.
- hash_seans – хеш-значення сеансу, має тип varchar(500), не може бути NULL.
- date – дата і час створення запису, має тип timestamp, не може бути NULL.

3.2. Розробка користувацького інтерфейсу для користувача із роллю куратора

Користувацький інтерфейс куратора є однією з ключових частин інформаційної системи для обліку та контролю виховної роботи зі студентськими групами. [42] Інтерфейс розроблено з урахуванням специфіки роботи кураторів у вищих навчальних закладах та має на меті забезпечити максимальну ефективність та зручність при виконанні їхніх обов'язків.

При вході на сайт користувач стикається із вікном авторизації, зображеним на рисунку.

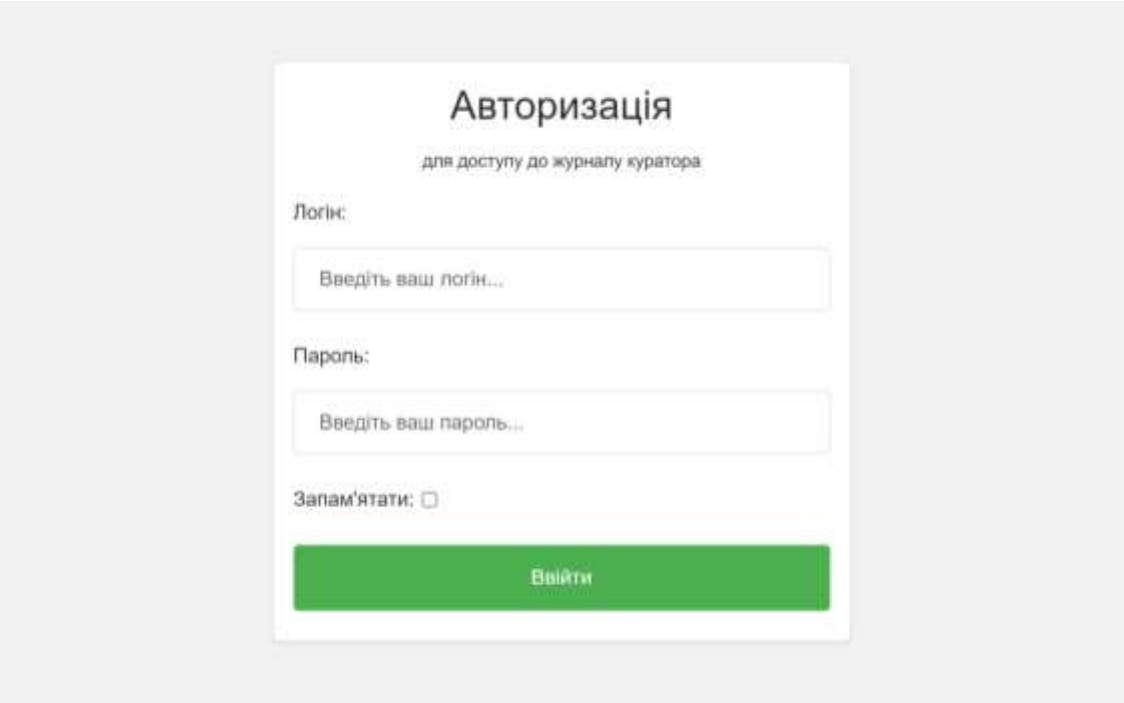


Рисунок 3.2 – Вікно входу в обліковий запис

Для авторизації в системі куратори використовують свої облікові дані для АСУ. При натисканні кнопки ввійти, система звертається до БД на сервері АСУ і отримує відповідь від нього. [43] Якщо ж збіг знайдено, то куратор потрапляє на головну панель сайту. Інакше система відобразить помилку про неправильний логін чи пароль.

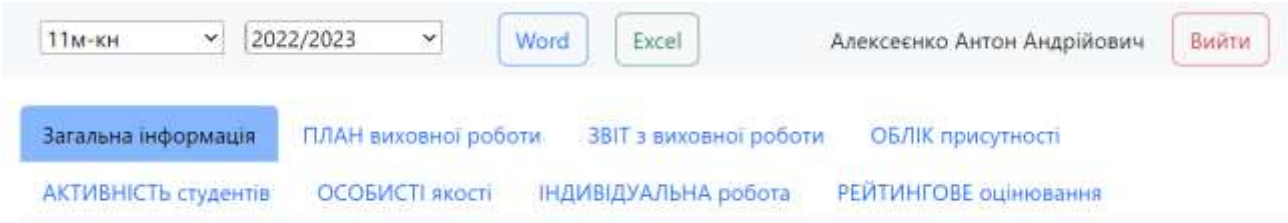


Рисунок 3.3 – Меню інформаційної системи

Меню інформаційної системи [44] завжди закріплено над областю сторінок журналу, пов’язаних із роботою куратора, та має наступні властивості: вибір групи, вибір періоду групи, формування звіту в файл Word чи Excel, вихід та навігація по основним блокам журналу куратора.

Після авторизації куратор потрапляє в першу чергу на сторінку "Загальна інформація", що містить персональні дані про студентів своєї групи, включаючи контактну інформацію, а також відомості про батьків. Куратор може додавати, редагувати або видаляти інформацію у цьому блоці, окрім полів ПІБ та дати народження, які заповнюються системою автоматично.

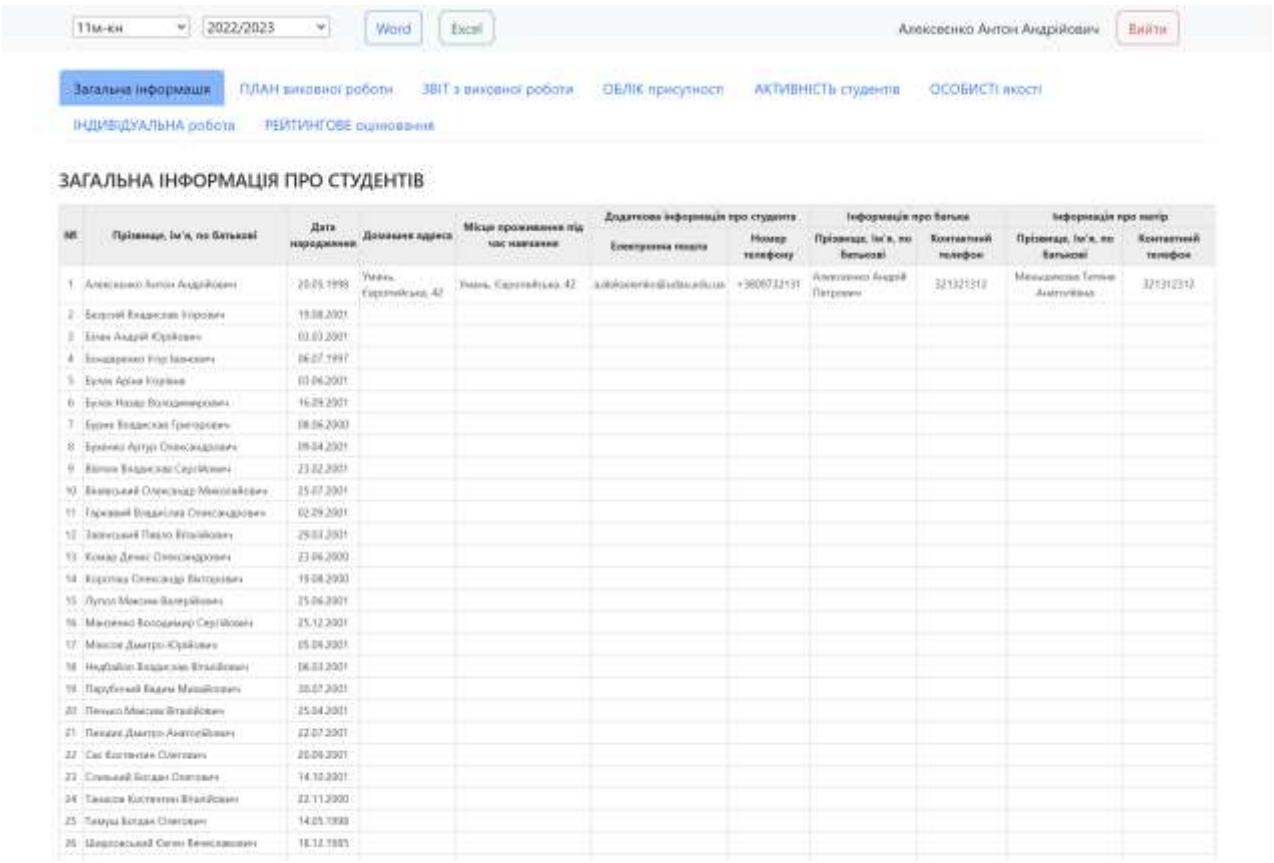


Рисунок 3.4 – Сторінка "Загальна інформація"

Перемкнувшись на сторінку "План виховної роботи", куратор має можливість здійснювати планування виховних заходів для своєї студентської групи. Ця сторінка є інтерактивною та дозволяє куратору вводити деталізовану інформацію про кожний запланований захід.

Куратор може додавати нові заходи, вказуючи їх назву, термін виконання та відповідальних за виконання, а також механізм перенесення заходів у звіт та їх видалення з журналу, що дозволяє систематизувати виховну роботу та забезпечує зручний механізм для її контролю.

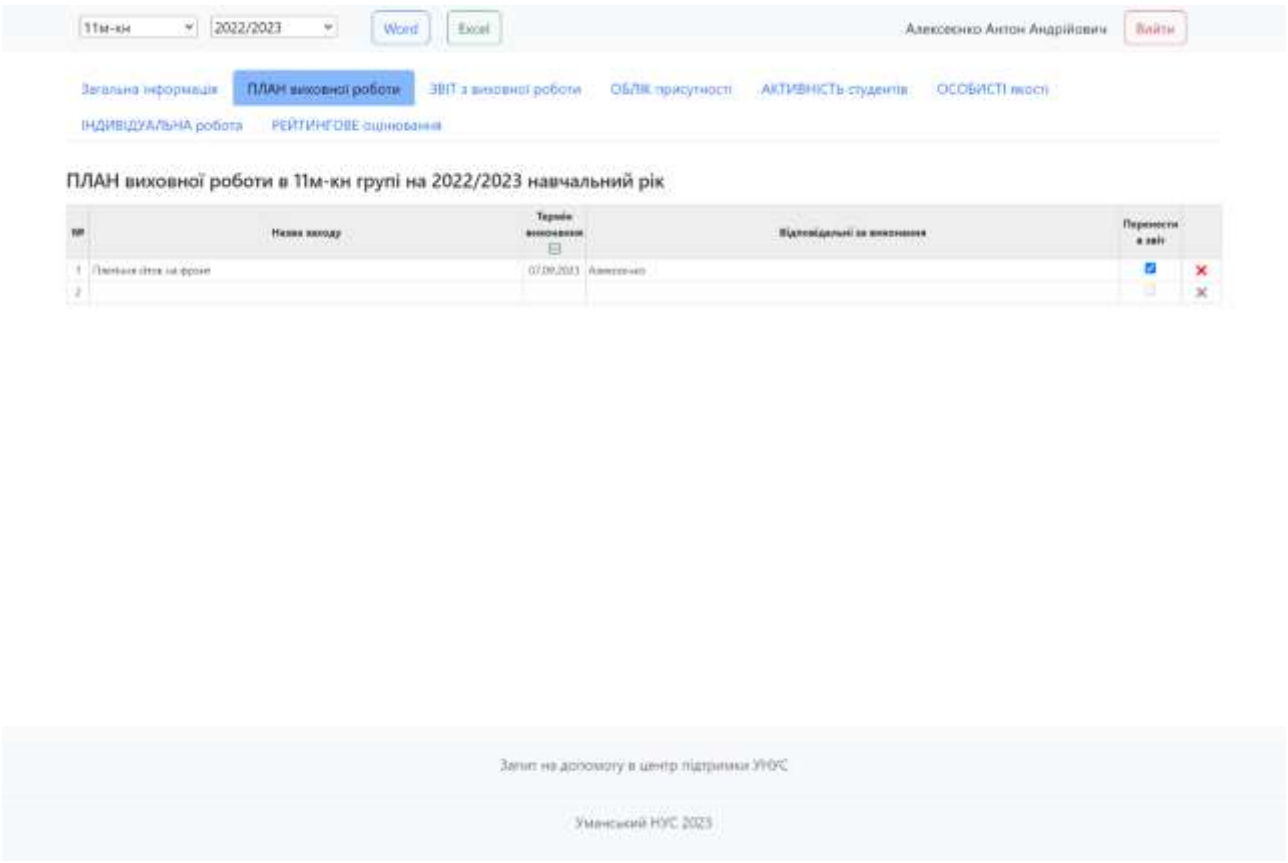


Рисунок 3.5 – Сторінка "План виховної роботи"

На сторінці "Звіт з виховної роботи," куратор має можливість вести облік всіх проведених виховних заходів. Ця сторінка є центральним місцем для зберігання та відображення даних про виховну діяльність, яка була здійснена зі студентською групою.

Куратор може додавати інформацію про проведені заходи, включаючи деталі про дату, місце проведення, кількість учасників, та інші важливі аспекти. Це дозволяє не тільки зберігати історію виховної роботи, але і аналізувати її ефективність.

Однією з ключових особливостей цієї сторінки є можливість вибору заходів з переліку загальноуніверситетських заходів. [45] Це полегшує процес заповнення звіту, оскільки куратор може просто вибрати захід зі списку, замість того, щоб вводити всю інформацію вручну.

Також куратор може вводити дані про присутніх на заході студентів. Ця інформація автоматично синхронізується з розділом "Облік присутності," що дозволяє мати цілісну картину участі студентів у виховній роботі.

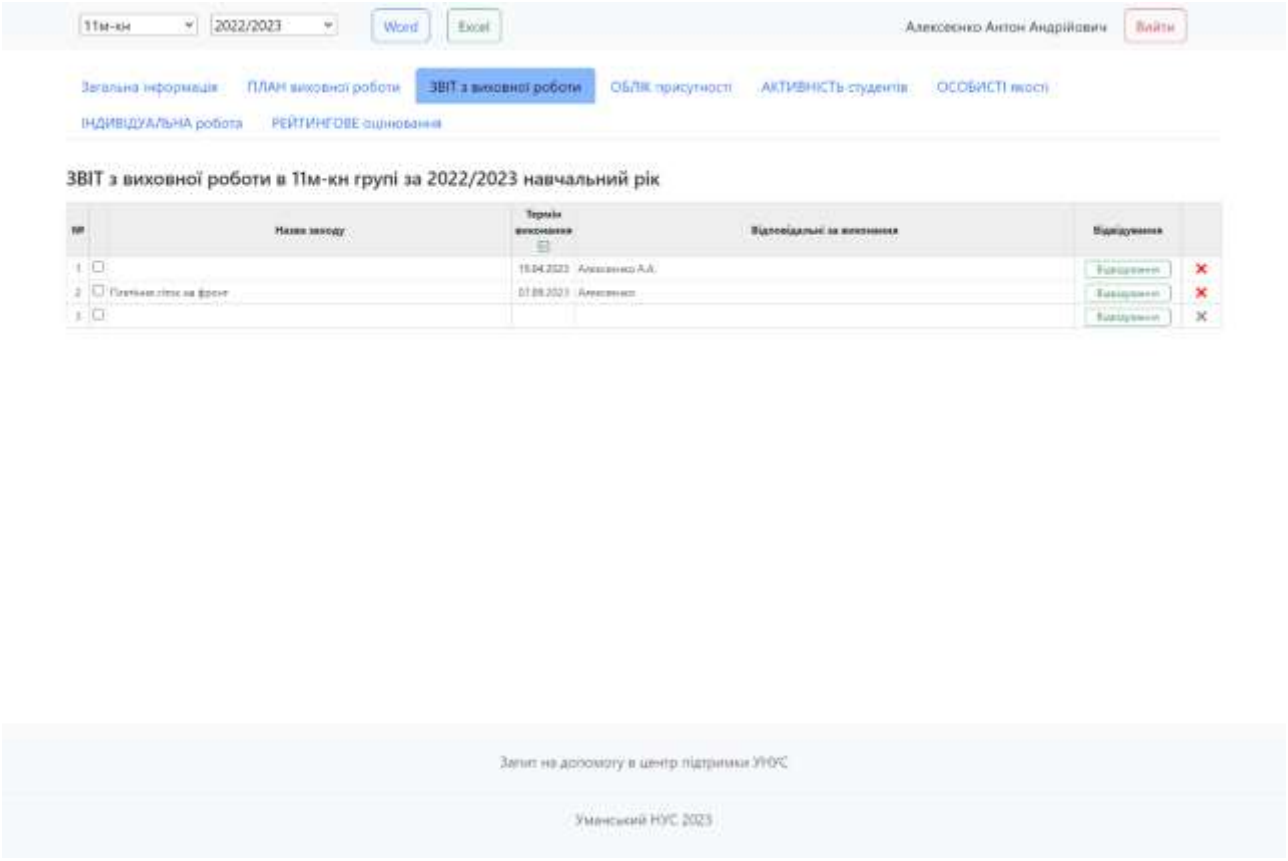


Рисунок 3.6 – Сторінка "Звіт з виховної роботи"

На сторінці "Облік присутності" куратор має можливість переглядати зібрану інформацію про участь студентів у різних виховних заходах. Ця сторінка є важливим інструментом для моніторингу активності студентів та їх взаємодії з виховним процесом.

Особливість цієї сторінки полягає в тому, що вся інформація щодо присутності автоматично синхронізується з розділом "Звіт з виховної роботи". Це означає, що при додаванні нового заходу або оновленні інформації про старий захід в розділі "Звіт з виховної роботи" дані про присутність автоматично оновлюються на цій сторінці. [46] Ця сторінка є нередагованою, що забезпечує об'єктивність і точність даних. Куратор не може змінювати

інформацію безпосередньо на цій сторінці, але може це зробити через розділ "Звіт з виховної роботи" з яким дані синхронізуються.

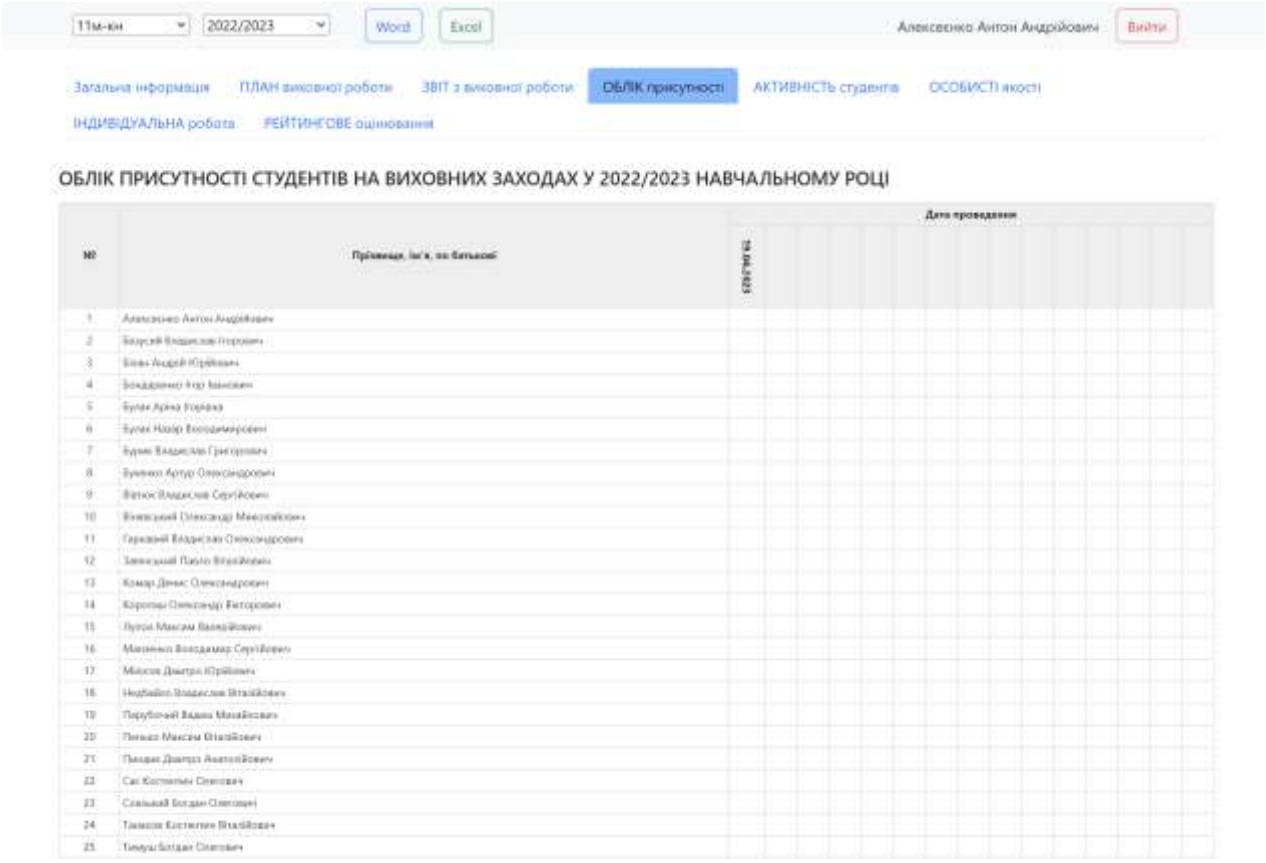


Рисунок 3.6 – Сторінка "Облік присутності"

Сторінка "Активність студентів" є центром для збору та аналізу інформації про участь студентів у різноманітних заходах. Куратор може додавати записи про активності студентів, вказуючи роль, функції та обов’язки студента в групі, роботу в клубах за інтересами, гуртках, спортивних секціях та участь у діяльності громадських організацій.

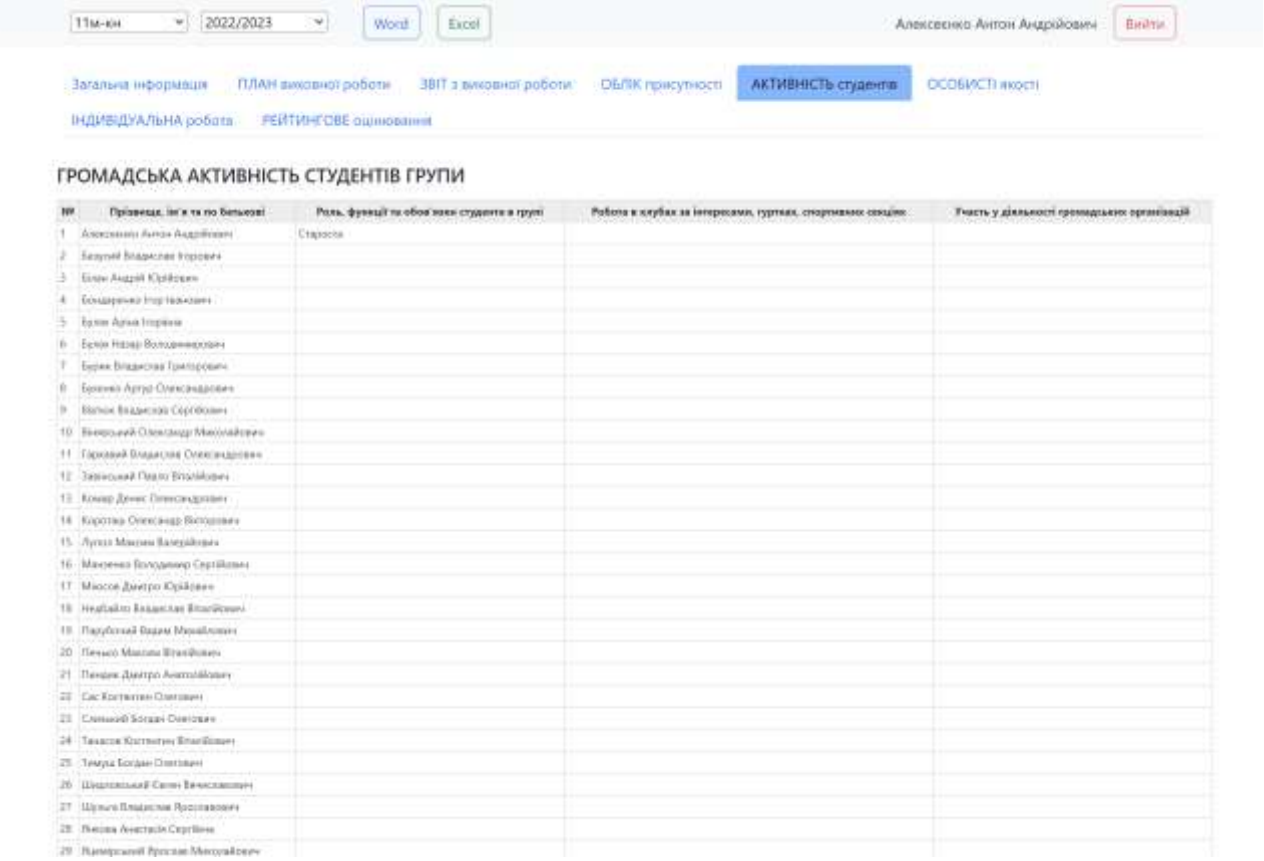


Рисунок 3.7 – Сторінка "Активність студентів"

На сторінці "Особисті якості" куратор має можливість здійснювати вибір особистих якостей студентів зі спеціально розробленого списку, який згрупований за категоріями, такими як "Загальні якості", "Моральні якості" та "Особисті якості". Це дозволяє куратору не лише систематизувати інформацію, але і визначити ключові аспекти, на які варто звернути увагу в процесі виховної роботи. [47]

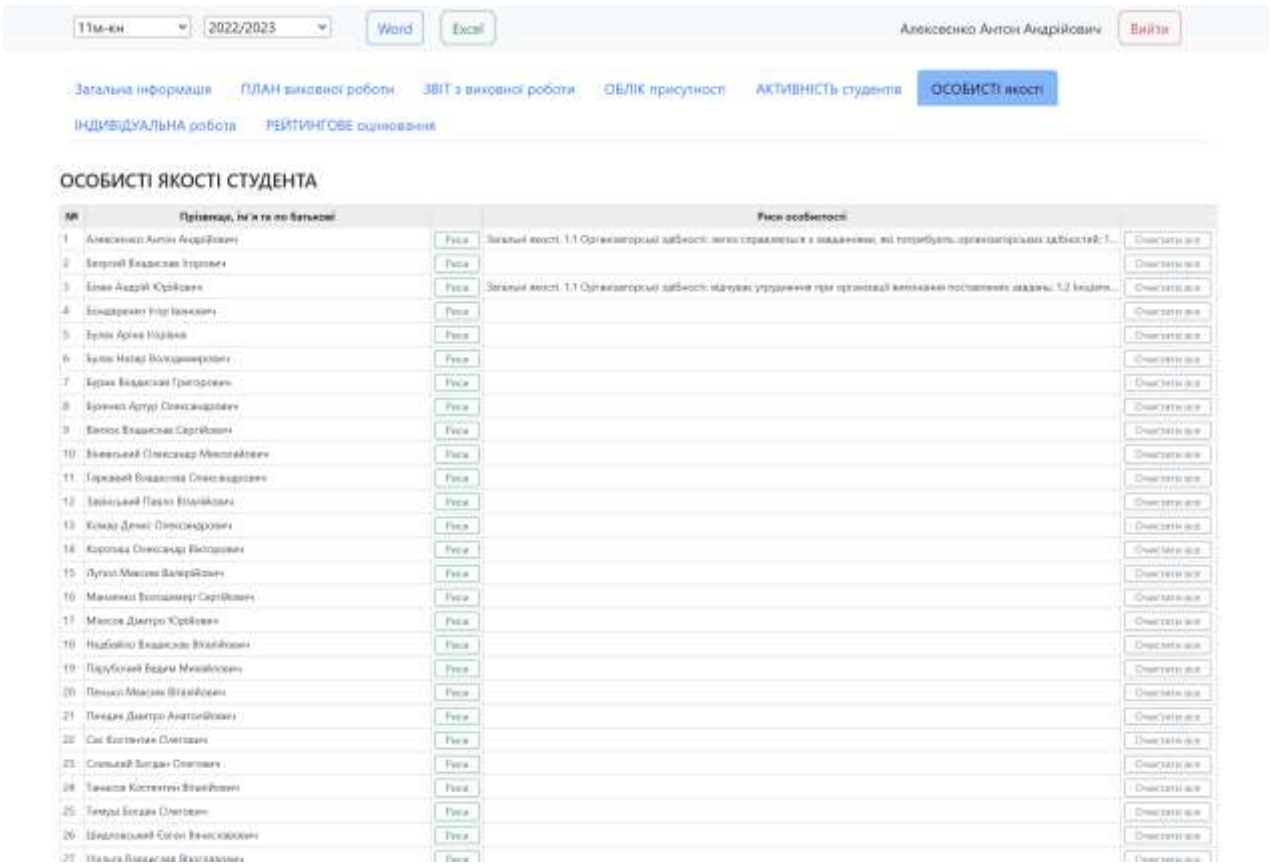


Рисунок 3.7 – Сторінка "Особисті якості"

Куратор має можливість детально оцінити особисті риси кожного студента. Поруч з ПІБ студента розташована кнопка "Риси", натискання на яку активує спливаюче вікно. У цьому вікні представлені різноманітні характеристики, які дозволяють куратору вибирати та встановлювати конкретні особисті риси студента.

Характеристики в меню поділені на категорії: "Загальні якості", "Моральні якості" та "Особисті якості", і кожна категорія містить підпункти, які детально описують конкретні риси особистості. Наприклад, у категорії "Загальні якості" можна вибрати підпункти, такі як "Організаторські здібності", "Ініціатива" та "Наполегливість", і для кожного з них визначити рівень прояву цих якостей у студента.

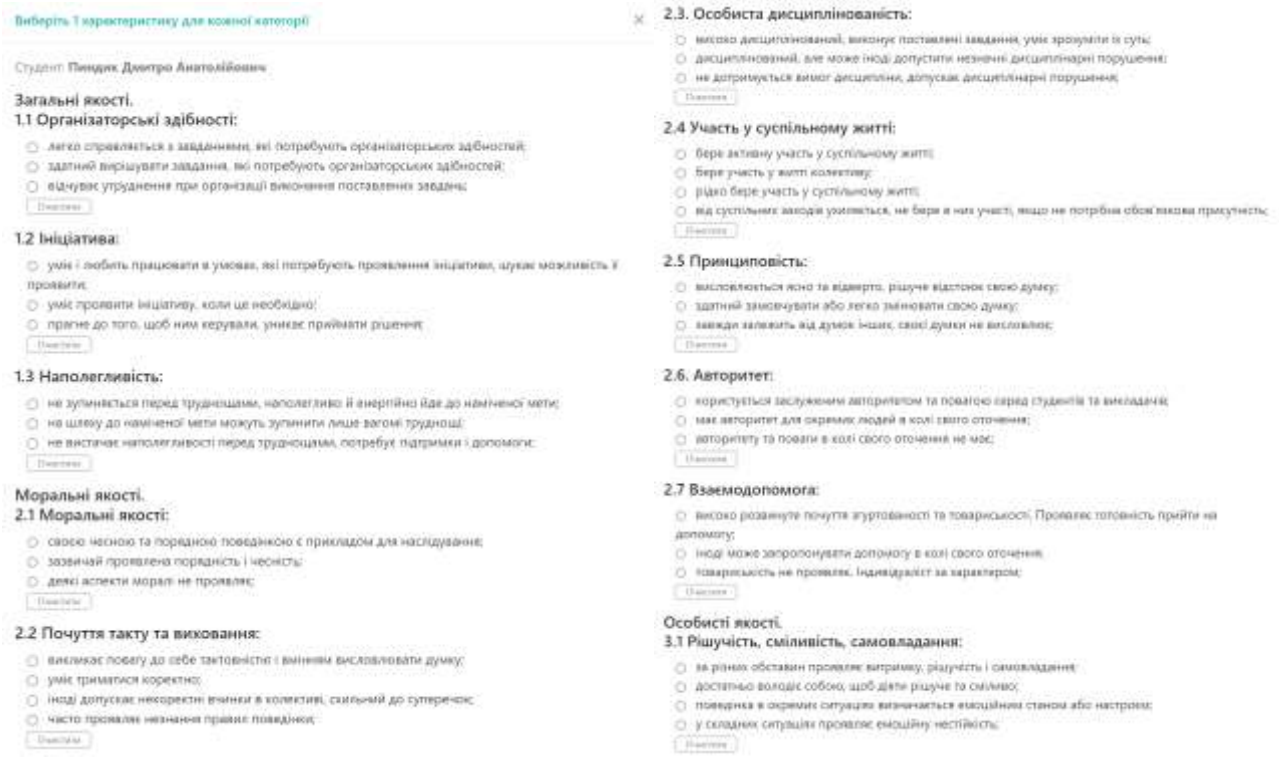


Рисунок 3.8 – Меню характеристики студента на сторінці "Особисті якості"

На сторінці "Індивідуальна робота" куратор може вести облік індивідуальної роботи, проведеної зі студентами. Таблиця на цій сторінці структурована таким чином, щоб надавати куратору зручний інтерфейс для ведення записів про кожну сесію індивідуальної роботи.

На цій сторінці куратор обирає дату проведення виховної роботи, ПІБ студента із каскадного списку, описує зміст виконаної індивідуальної роботи та вказує тривалість індивідуальної роботи в годинах.

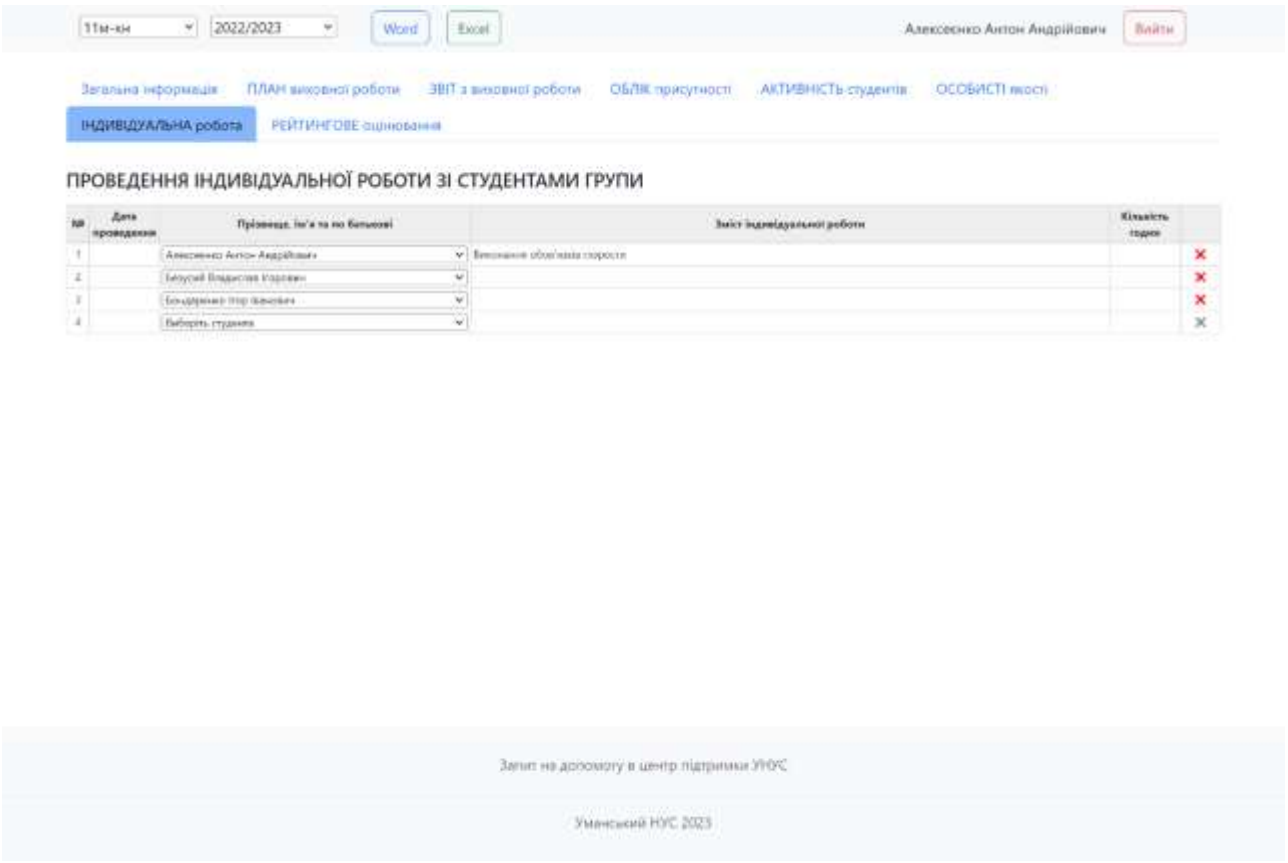


Рисунок 3.8 – Сторінка "Індивідуальна робота"

Сторінка "Рейтингове оцінювання" призначена для того, щоб куратор міг оцінювати студентів за різними категоріями діяльності. Це дозволяє куратору мати збалансований погляд на загальний розвиток та вклад кожного студента у життя групи та університету. [48]

На цій сторінці представлені категорії, які оцінюються за 5-бальною шкалою, де 5 є найвищим балом. Рейтингове оцінювання здійснюється за наступними категоріями:

- Наукова робота
- Громадська робота
- Спортивна діяльність
- Культурно-масова діяльність
- Соціальна активність (пасивність)
- Загальний результат

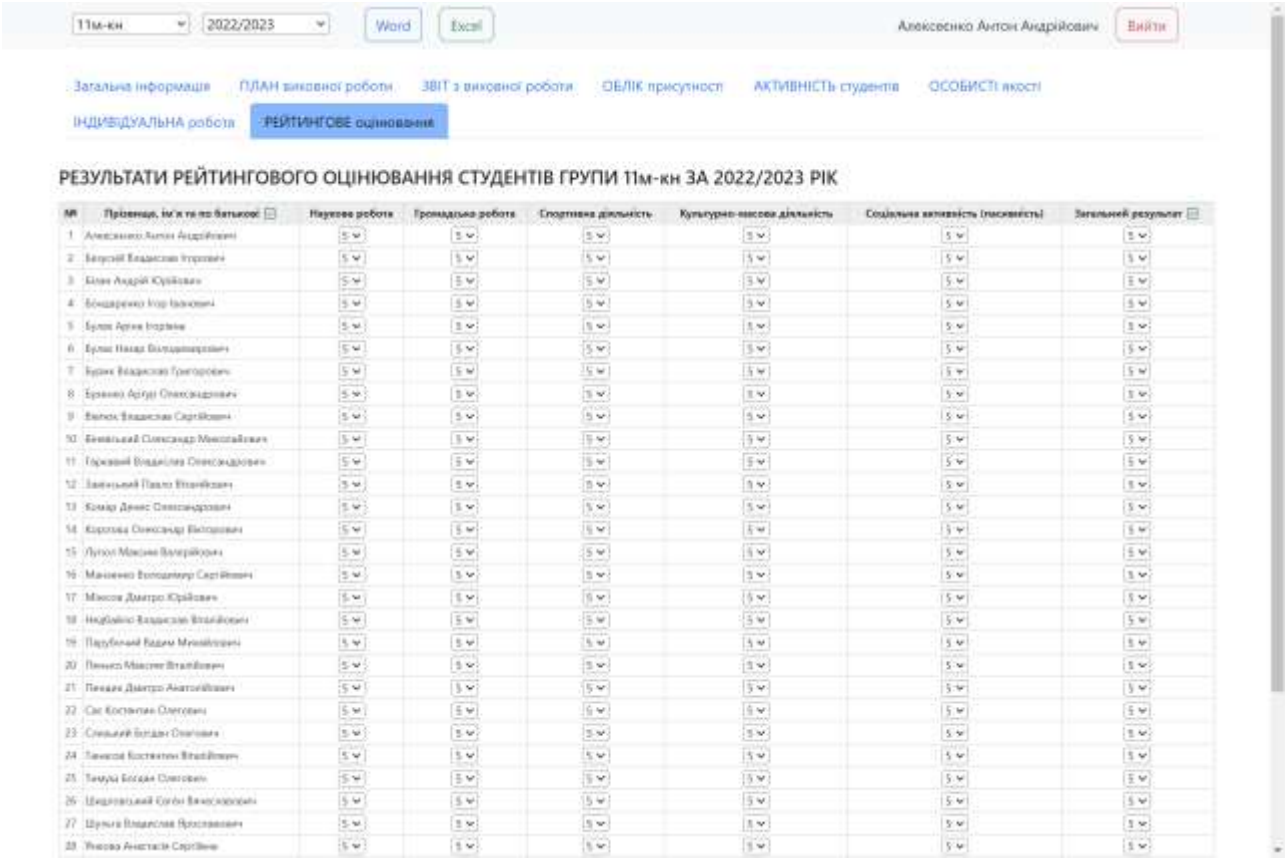


Рисунок 3.9 – Сторінка "Рейтингове оцінювання"

3.3. Розробка користувацького інтерфейсу адміністратора

Розробка користувацького інтерфейсу адміністратора є важливим етапом у створенні інформаційної системи, оскільки адміністратор відіграє ключову роль в керуванні системи. Адміністратор повинен мати функціонал для перегляду існуючих журналів кураторів, керувати списком заходів університету та переглядати відвідуваність. [49] Саме тому було розроблено панель керування для адміністратора сайту (рис 3.10 – 3.13).

На головній панелі відображається агрегована статистика по кураторам, що включає кількість кураторів, їхню активність та взаємодію зі студентами. Це дозволяє адміністратору відслідковувати ефективність роботи кураторів та їхній вклад у виховний процес.

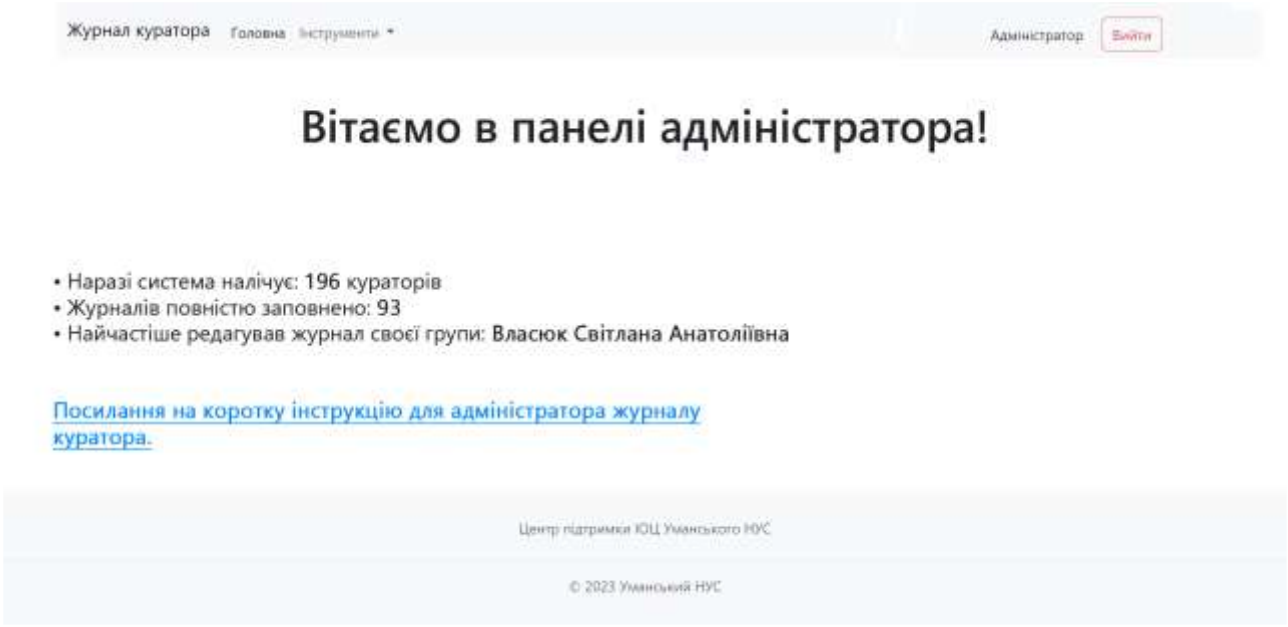


Рисунок 3.10 – Головна сторінка панелі адміністратора містить статистику

Сторінка із загальноуніверситетськими заходами надає адміністратору можливість додавати, редагувати та видаляти інформацію про заходи, що проводяться в університеті, та контролювати їхнє планування та проведення.

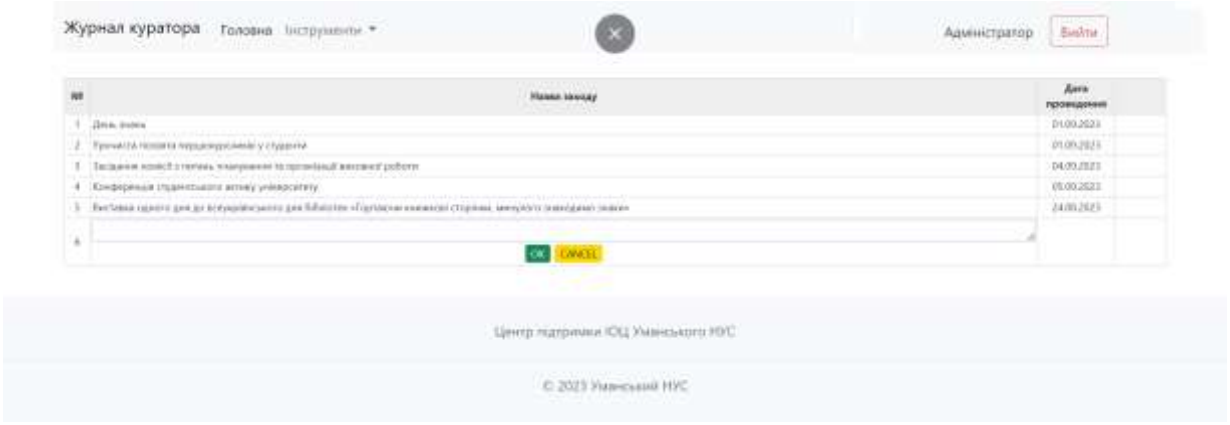


Рисунок 3.11 – Сторінка із загальноуніверситетськими заходами

Сторінка із вибором куратора для перегляду журналу в панелі адміністратора дозволяє адміністратору вибирати конкретного куратора та переглядати всі журнали, які він веде, що забезпечує моніторинг виховної роботи, проведеної куратором із студентськими групами.

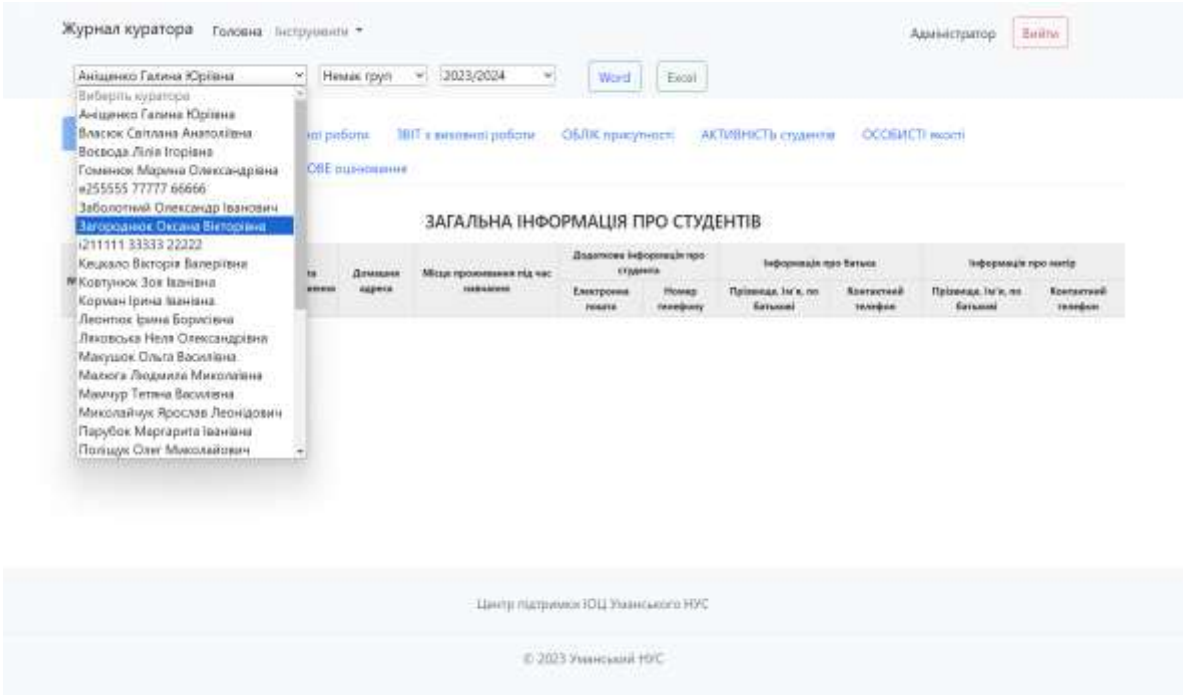


Рисунок 3.12 – Сторінка із вибором куратора для перегляду журналу

Сторінка для відслідковування відвідуваності кураторами дозволяє адміністратору відслідковувати активність кураторів у системі. Ця сторінка зосереджена на відображенні останніх 10 візитів кураторів, надаючи адміністратору оперативну інформацію про час та дату візиту, а також інші важливі параметри відвідуваності. [50]

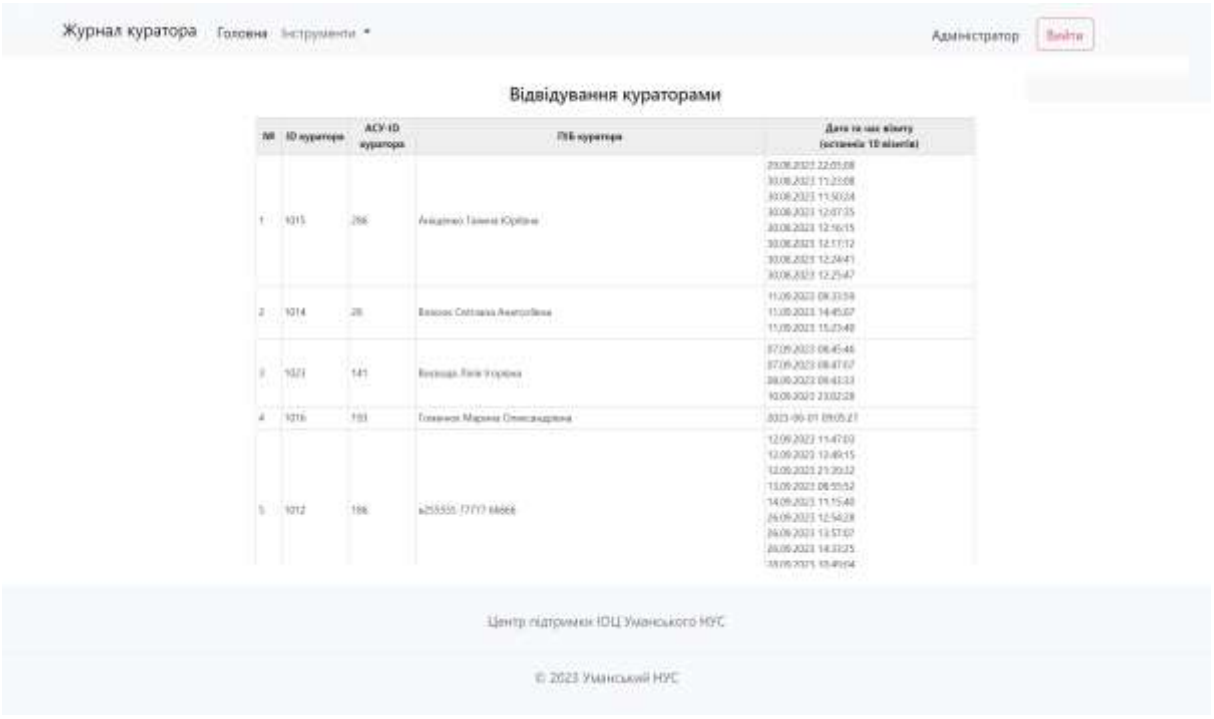


Рисунок 3.13 – Сторінка для відслідковування відвідуваності кураторами

3.4. Реалізація функціоналу збору та аналізу звітної інформації

Розробка надійної системи збору та аналізу даних в середовищі університету є важливою ініціативою, спрямованою на підвищення якості та результативності освітньої роботи. Інтеграція системи з передовими інструментами, такими як PHPOffice, завдяки яким з'являється здатність створювати вичерпні звіти у форматах Word та Excel. Ці функції гарантують, що система здатна не лише збирати широкий спектр даних, але й представляти їх у форматах, корисних відповідальним особам з виховної роботи в оцінці ефективності цієї роботи. [52]



Рисунок 3.14 – Функціональні кнопки для створення звітів у відповідних форматах Word та Excel

Функція експорту в Excel дозволяє створювати комплексне табличне представлення електронного журналу куратора. Цей детальний формат сприяє поглибленому аналізу та веденню обліку, дублюючи повний спектр даних, доступних у веб-версії системи. Функція експорту до Word, з іншого боку, призначена для створення формальних документів, готових до друку та подальшого фізичного архівування на кафедри. На рисунках представлений зовнішній вигляд системи у табличному варіанті Excel (рис 3.15 – 3.16).

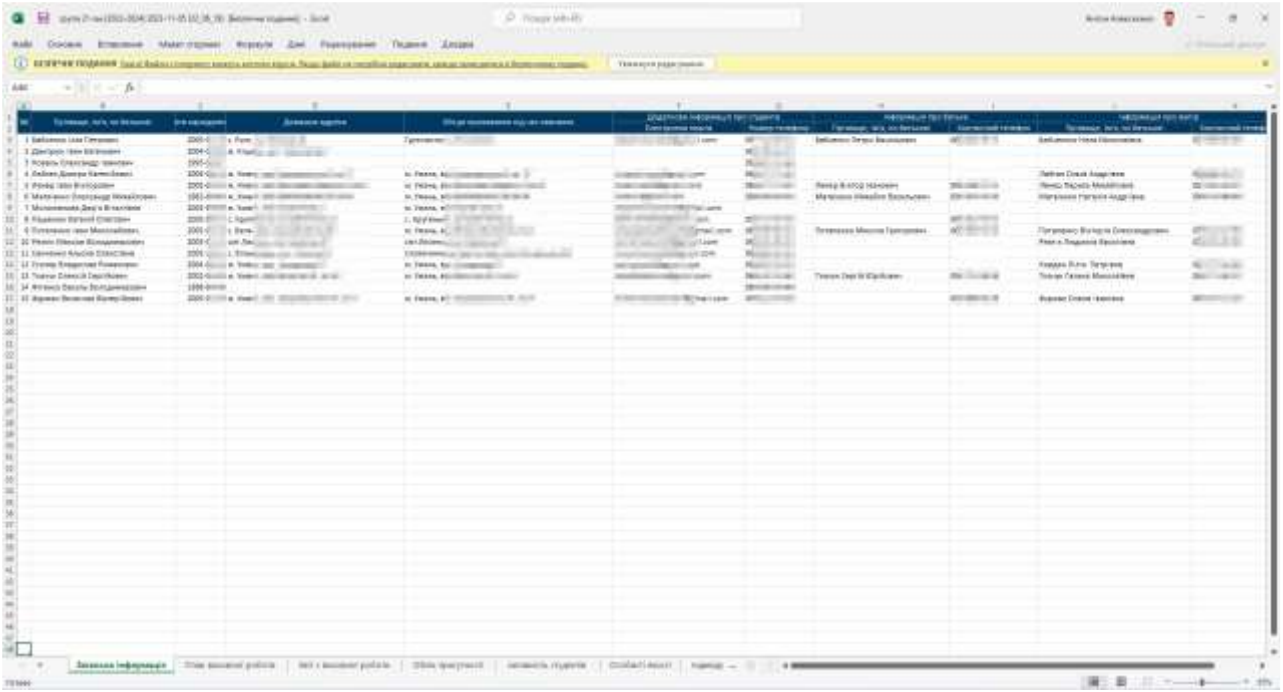


Рисунок 3.15 – Лист "Загальна інформація" у табличному варіанті Excel

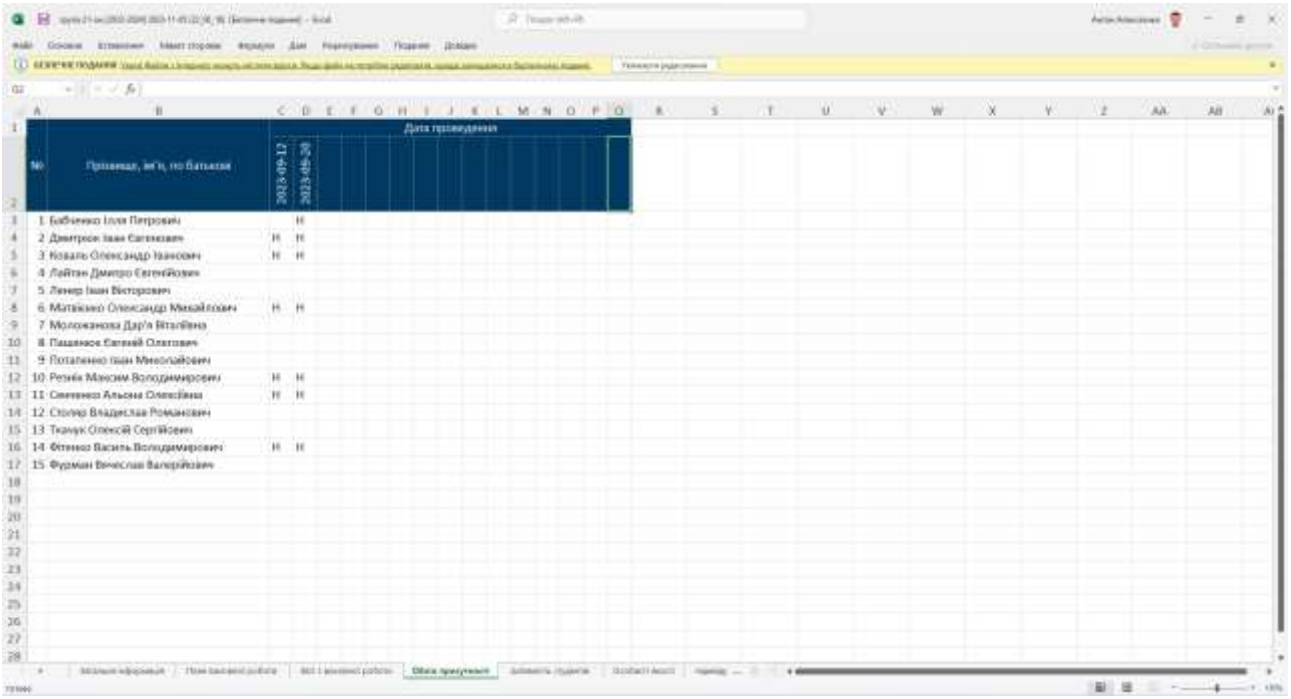


Рисунок 3.16 – Лист "Облік присутності" у табличному варіанті Excel

Наприкінці навчального року журнал куратора роздруковується та зберігається на кафедрі, що забезпечує збереження історії виховної роботи та можливість її аналізу в майбутньому. [51] Це також відповідає нормативним документам, що регламентують виховну діяльність в університеті, забезпечуючи відповідність роботи кураторів встановленим стандартам і

процедурам. На рисунках представлений зовнішній вигляд системи у форматі Word (рис 3.17 – 3.22).

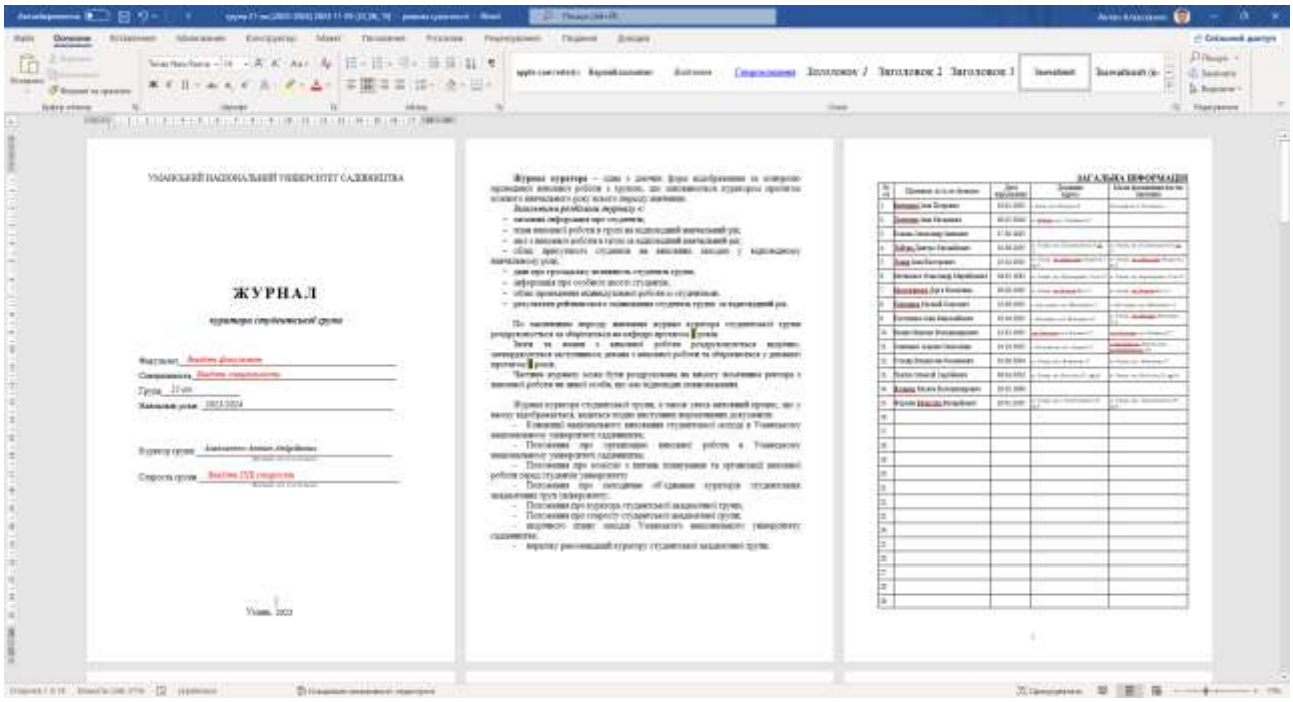


Рисунок 3.17 – Представлення інформаційної системи у форматі Word

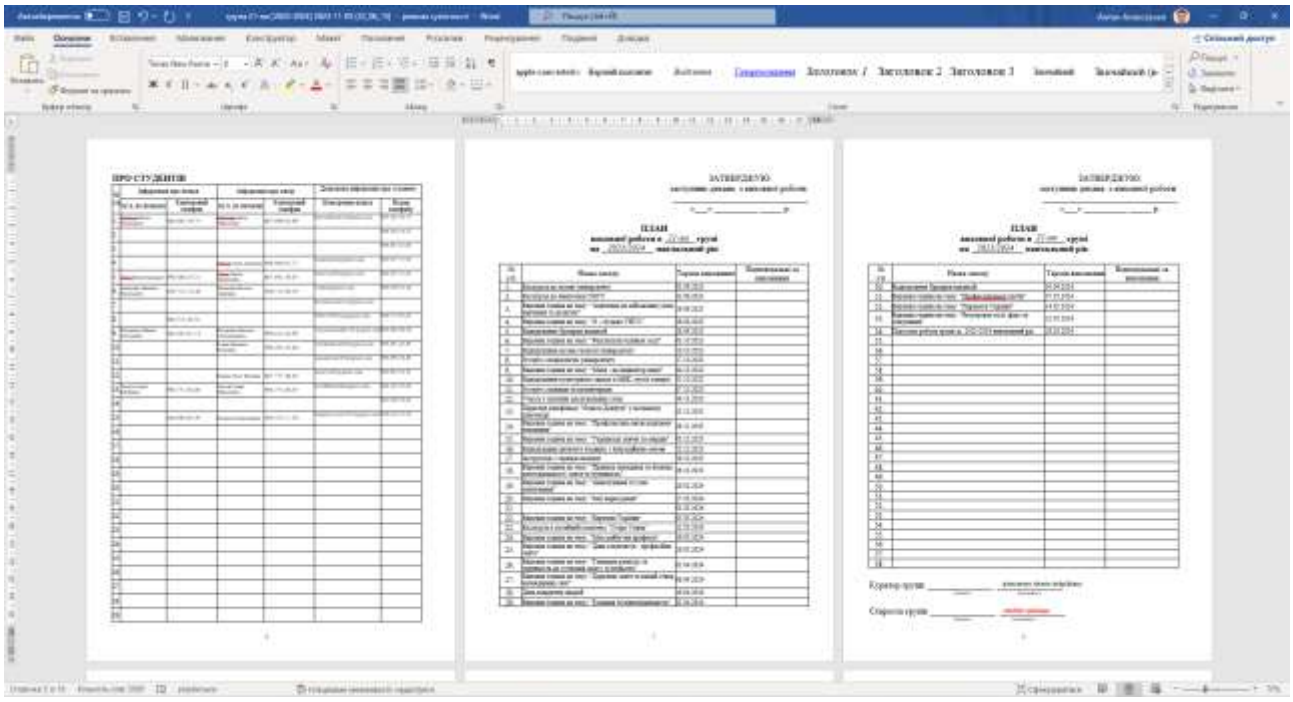


Рисунок 3.18 – Представлення інформаційної системи у форматі Word

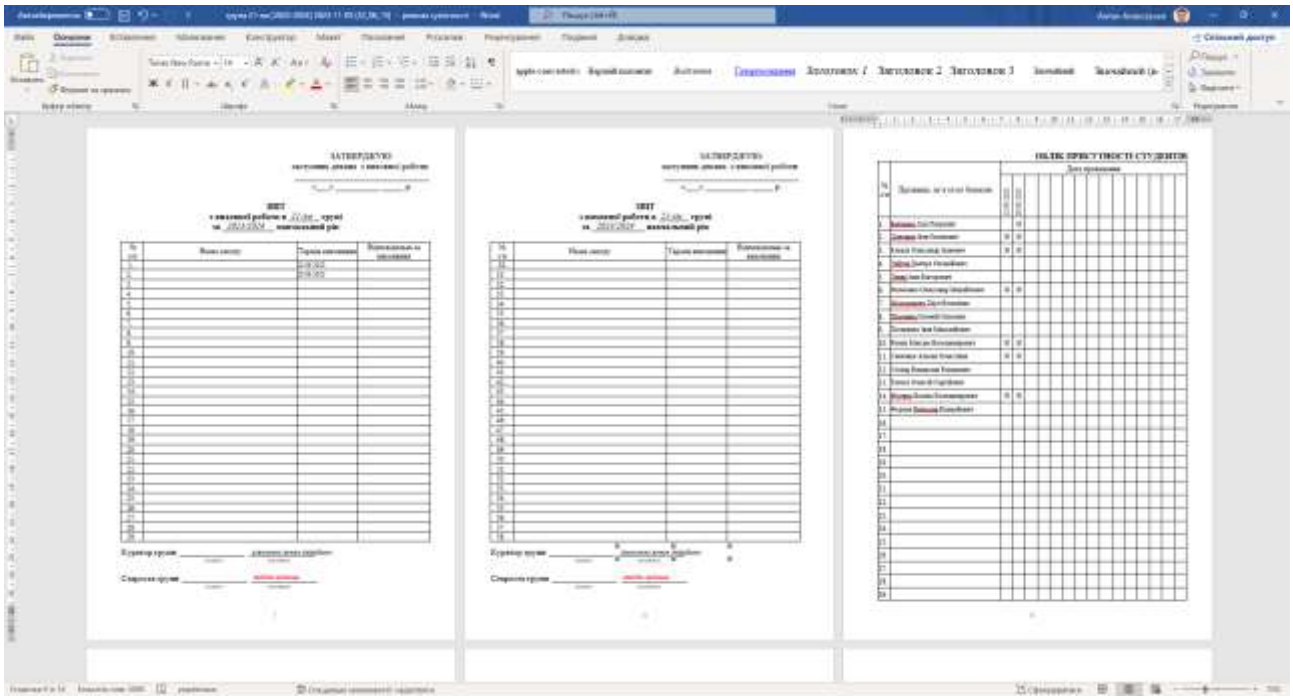


Рисунок 3.19 – Представлення інформаційної системи у форматі Word

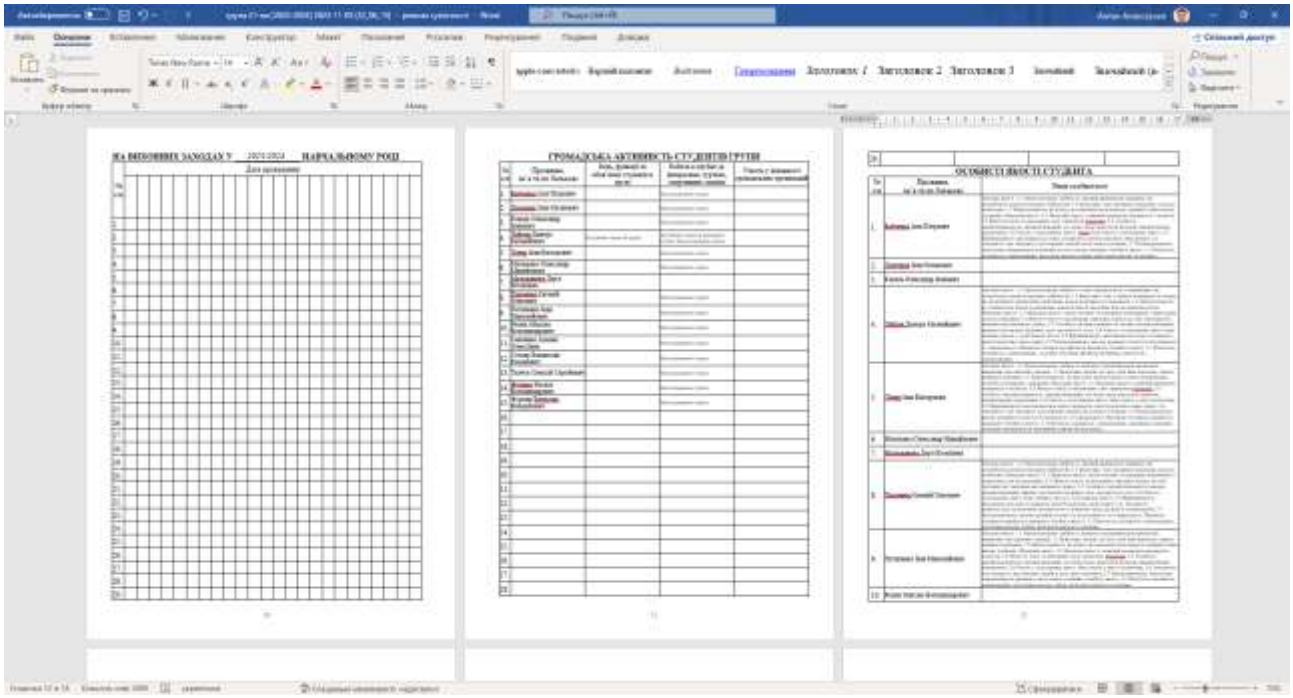


Рисунок 3.20 – Представлення інформаційної системи у форматі Word

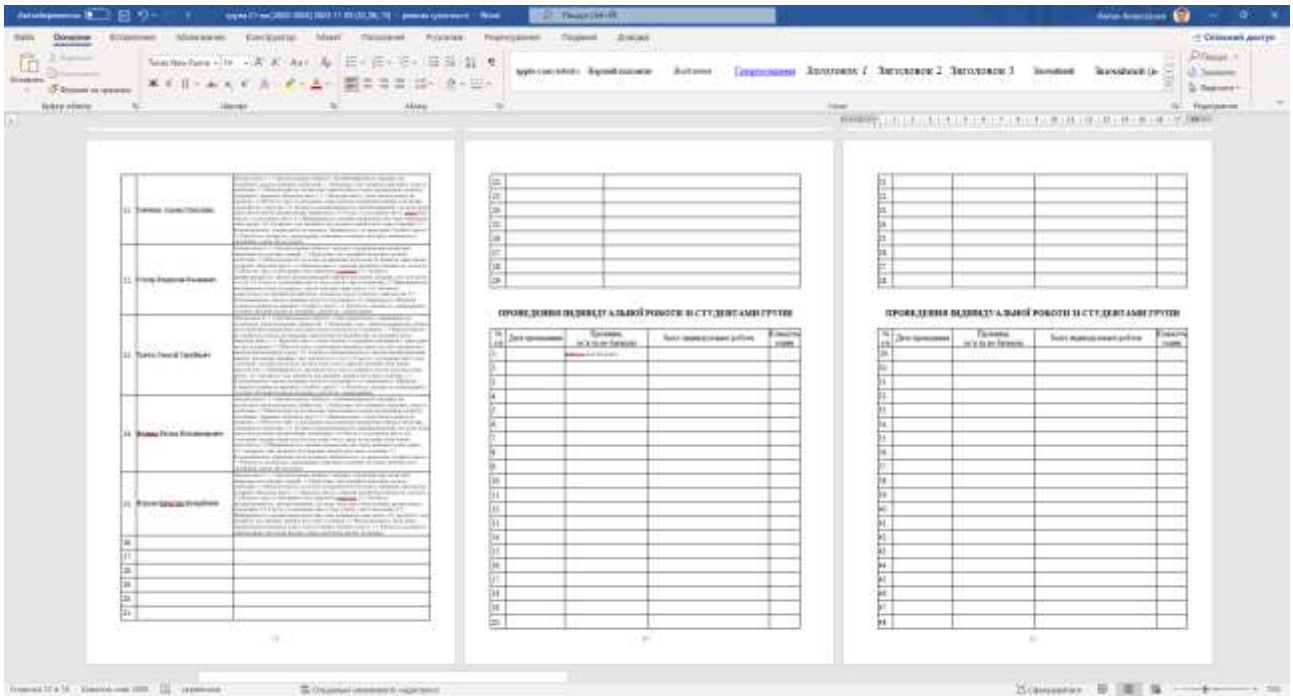


Рисунок 3.21 – Представлення інформаційної системи у форматі Word

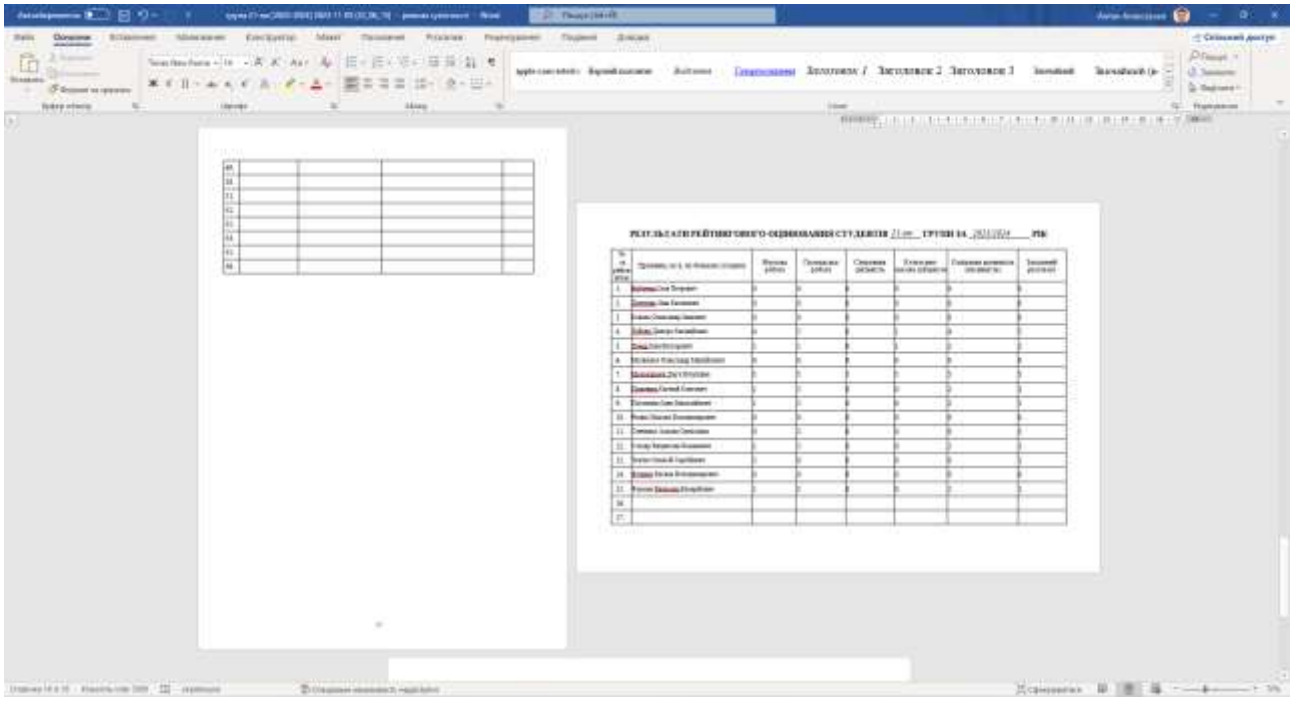


Рисунок 3.22 – Представлення інформаційної системи у форматі Word

Таким чином, реалізація функціоналу зі збору та аналізу звітної інформації є ключовим елементом підтримки високої якості виховної роботи, що дозволяє відповідальним особам точно оцінювати ефективність цієї діяльності. Можливість системи генерувати звіти та плани у форматах Word та

Excel не лише спрощує цей процес, але й забезпечує відповідність документації освітнім стандартам та методичним рекомендаціям університету.

ВИСНОВКИ

У процесі вивчення та аналізу проблеми обліку та контролю виховної роботи з студентськими групами було виявлено значну необхідність автоматизації цього процесу. Це дозволило підвищити ефективність та зручність роботи кураторів, забезпечивши їм можливість зосередитись на індивідуальному підході до кожного студента.

Вибір технологій та архітектури системи був здійснений з урахуванням необхідності гнучкості, масштабованості та легкості управління кодом. Розроблена інформаційна система забезпечує кураторам зручний інтерфейс для обліку та контролю виховної роботи. Система дозволяє кураторам планувати, звітувати, вести облік присутності, оцінювати активність та особисті якості студентів. Це сприяє всебічному розвитку студентів та підвищенню ефективності виховної роботи.

Успішна інтеграція системи з АСУ дозволяє синхронізувати дані про кураторів та їх групи та забезпечує авторизацію користувачів через існуючу систему. Це забезпечує цілісність та актуальність даних, що використовуються у системі.

Розроблена система має великий потенціал для подальшого розвитку та удосконалення. Можливе додавання нових функцій, інтеграція з іншими системами та адаптація до змінюваних вимог та потреб користувачів.

Впровадження системи вже показало свою ефективність, сприяючи підвищенню якості та ефективності виховної роботи. Куратори отримали можливість зосередитись на розвитку компетенцій та особистісних якостей студентів, що є ключовим для їхнього професійного та особистого розвитку.

Загалом, розроблена інформаційна система виявилася успішною та ефективною у вирішенні поставлених завдань та відповідає всім вимогам та очікуванням користувачів.

СПИСОК ЛІТЕРАТУРИ

1. Реформа освіти та науки. [Електронний ресурс] – Режим доступу: https://www.kmu.gov.ua/diyalnist/reformi/rozvitok-lyudskogo-kapitalu/reforma-osviti
2. Черниш О.В. Стратегічне управління освітньою діяльністю у ЗВО. – Економіка і суспільство. 2018. № 19. – 689 с.
3. Мала, І. Б. Дистанційне навчання як дієвий інструмент управлінської освіти. Вчені записки Університету "КРОК". – Київ: Університет економіки та права "КРОК", 2022 – 132-151 с.
4. Лузік Е.В., Демченко Н.І., Проскурка Н.М. Світові моделі університетської освіти в системі дистанційного навчання. Серія: Педагогіка. Психологія: зб. наук. пр. – Київ. : Національний авіаційний університет, 2020. – Вип. 2(17). – 47 с.
5. Середа, І. Використання технологій корекційно-виховної роботи у практичній діяльності спеціальних педагогів. – АКТУАЛЬНІ ПИТАННЯ ГУМАНІТАРНИХ НАУК Дрогобич: Видавничий дім "Гельветика", 2020. – Вип. 27. Том 4. – 284 с.
6. Edusys [Електронний ресурс] – Режим доступу: https://www.edusys.co/
7. SchoolSoft [Електронний ресурс] – Режим доступу: https://www.schoolsoft.se/
8. Moodle [Електронний ресурс] – Режим доступу: https://moodle.org/
9. Athaya, H., Nadir, R. D. A., Indra Sensuse, D., Kautsarina, K., Suryono, R. R. Moodle Implementation for E-Learning: A Systematic Review. – In Proceedings of the 6th International Conference on Sustainable Information Engineering and Technology, 2021 – 106-112 с.
10. Автоматизована система управління навчальним процесом. [Електронний ресурс] – Режим доступу: https://mkr.org.ua/
11. Удосконалення освітньо-виховного процесу в закладі вищої освіти: збірник науково-методичних праць – Мелітополь: ТДАТУ, 2022. – Вип. 25. – 348 с.

12. Веб-технології.: підручник / Львів : "Магнолія-2006", 2018. – 336 с
13. Glass, E. How to build a website with HTML – New York: DigitalOcean, 2020 – 172 с.
14. Learn to Code HTML & CSS [Електронний ресурс] – Режим доступу до ресурсу: https://learn.shayhowe.com/html-css/
15. ROSTISLAV, M., & OLEKSANDR, H. ВЕБ-ТЕХНОЛОГІЇ ТА ДИЗАЙН. – Хмельницький: ХНУ, 2021. – 22 с.
16. HTML CSS JavaScript PHP SQL. W3Schools українською. Уроки для початківців. [Електронний ресурс] – Режим доступу: https://w3schoolsua.github.io/#gsc.tab=0
17. Bootstrap · The most popular HTML, CSS, and JS library in the world. [Електронний ресурс] – Режим доступу: https://getbootstrap.com/
18. Bublyk. A. Web page design as a basis for readability, content and aesthetics of a modern website. – Lviv: Computer Academy IT STEP University, 2021 – 5-11 с.
19. Mardan, A., Mardan, & Corrigan. Practical Node. js. – Apress, 2018 – 23 с.
20. Доскоч, Н., & Цуприк, Г. Б. Розробка системи забезпечення безпечного обміну інформацією з використаннм технологій веб програмування. Матеріали X науково-технічної конференції "Інформаційні моделі, системи та технологі" – Тернопіль: ТНТУ імені Івана Пулюя, 2022 – 114-114 с.
21. Hong, P. Practical web design: Learn the fundamentals of web design with HTML5, CSS3, bootstrap, jQuery, and vue. js. Packt Publishing Ltd. – Packt Publishing, 2018 – 260 с.
22. Ceklosky, S. Client side web programming more or less: AJAX or JQUERY. – Journal of Computing Sciences in Colleges, 2018 – 63-64с.
23. Nixon R. Learning PHP, MySQL & JavaScript / Robin Nixon – 5th ed. – Sebastopol: O'Reilly, 2018. – 832 с.

24. Kevin Tatroe Programming PHP, 4th Edition / Kevin Tatroe – 4th ed. – Sebastopol: O'Reilly, 2020. – 544 с.
25. PHP: The Right Way [Електронний ресурс] – Режим доступу до ресурсу: https://phptherightway.com/.
26. PHPOffice · GitHub. [Електронний ресурс] – Режим доступу до ресурсу: https://github.com/PHPOffice
27. Halupecki, E. The role of the middle layer in the web environment – Doctoral dissertation, Bjelovar University of Applied Sciences. Department of Computer Science, 2023 – 4 с.
28. Christudas, B., & Christudas, B. MySQL – Berkeley, CA: Apress, 2019 – 877-884 с.
29. Nixon, R. Learning PHP, MySQL & JavaScript. – O'Reilly Media, Inc., 2021 – 826 с.
30. phpMyAdmin. [Електронний ресурс] – Режим доступу до ресурсу: https://www.phpmyadmin.net/
31. Мулеса О.Ю., Варга Я.В. Інформаційні системи та реляційні бази даних / О.Ю. Мулеса, Я.В. Варга, навч. посібник. – Ужгород, 2023. – 132 с.
32. Visual Studio Code - Code Editing. Redefined. [Електронний ресурс] – Режим доступу до ресурсу: https://code.visualstudio.com/
33. Алгоритмізація та програмування / Ковалюк Т. В. – Львів: "Магнолія 2006", 2018. – 400 с.
34. Двірничук К. В., Вацек Д. О. Веб-програмування та веб-дизайн: навчальний посібник. – Чернівці : ЧНУ ім. Ю. Федьковича, 2022. – 472 с.
35. Ломачинська, І. М., Ломачинський, Б. Г. Роль інформаційної культури у регулюванні соціальних інформаційних систем. – Львів: Вісник Львівського університету, 2020 – 94 с.
36. Choi, K., Chang, B. M. A theory of RPC calculi for client–server model. – Cambridge: Journal of Functional Programming, 2019 – 29 с.

37. Берко А.Ю., Верес О.М., Пасічник В.В. Системи баз даних та знань. Книга 2 – Львів : "Магнолія-2006", 2018. – 584 с.
38. Meier, A., Kaufmann, M. SQL & NoSQL databases. Berlin/Heidelberg: Springer Fachmedien Wiesbaden, 2019 – 163 с.
39. Yunus, M. A. M., Brohan, M. Z., Nawi, N. M., Surin, E. S. M., Najib, N. A. M., Liang, C. W. Review of SQL injection: Problems and prevention. – JOIV: International Journal on Informatics Visualization, 2018 – 215-219 с.
40. Stair, R. M., Reynolds, G. W. Fundamentals of information systems. Cengage Learning, 2018. – 296 с.
41. Haverbeke M. Eloquent JavaScript: A Modern Introduction to Programming / Marijn Haverbeke. – 3rd ed. – San Francisco: No Starch Press, 2018. – 463 с.
42. Шаховська Н.Б., Литвин В.В. Проектування інформаційних систем. Навчальний посібник – Львів: "Магнолія-2006", 2018. – 380 с.
43. Siriwardena, P. Advanced API security: OAuth 2.0 and beyond – CA, USA: Apress, 2020 – 103-127 с.
44. В. В. Пасічник, О. В. Пасічник. Веб-дизайн: підручник – Львів: "Магнолія-2006", 2018. – 518 с.
45. Алмашій, І. І. Соціально-психологічні характеристики екологічної складової загальноуніверситетських семінарів. – Мукачево: МДУ, 2020 – Випуск 1(11) – 274-276 с.
46. Полив'яна О. А. Сучасні автоматизовані системи управління закладом вищої освіти. – Полтава: Кафедра загальної педагогіки та андрагогіки ПНПУ імені В. Г. Короленка, 2021. – 299–302 с.
47. Шахова, О. Г. Соціальна психологія особистості: навчальний посібник – Харків: Контраст, 2019. – 116 с.
48. Ісаєнко, Ю. В., Горбунова, Н. І. Рейтингова система оцінювання навчальних досягнень студентів. – Матеріали Всеукраїнської дистанційної науково-методичної конференції педагогічних

працівників закладів вищої освіти – Харків: Коледж НФаУ, 2019. – 191 с.

49. Пастернак І.І., Костик А.Т. Інструментальні засоби веб-технологій – Львів: "Магнолія-2006", 2023. – 198 с.
50. Юрчак І. Ю. Базові засади веб-розробки. Лабораторний практикум з дисципліни "Глобальні інформаційні мережі": навчальний посібник – Львів: Магнолія 2006, 2022. – 180 с.
51. Положення про забезпечення якості освітньої діяльності та якості вищої освіти в Уманському НУС [Електронний ресурс] – Режим доступу до ресурсу: https://www.udau.edu.ua/ua/file/85IT
52. Kromann, F. Beginning PHP and MySQL: from novice to professional. – Apress, 2018 – 824 с.

ДОДАТОК А
ЛІСТИНГ КОДУ ІНФОРМАЦІЙНОЇ СИСТЕМИ

```
<?php
include __DIR__ . "/../../app/tables/Table01.php";
include __DIR__ . "/../../app/tables/Table02.php";
include __DIR__ . "/../../app/tables/Table03.php";
include __DIR__ . "/../../app/tables/Table04.php";
include __DIR__ . "/../../app/tables/Table05.php";
include __DIR__ . "/../../app/tables/Table06.php";
include __DIR__ . "/../../app/tables/Table07.php";
include __DIR__ . "/../../app/tables/Table08.php";

$teacher = new \app\Teacher($auth_id);
$teacherFIO = $teacher->getTeacherFIO();
$_SESSION['curator_fio'] = $teacherFIO;

if(isset($_SESSION['group'])){
   $currentGroup = $_SESSION['group'];
} else {
   $currentGroup = $teacher->getDefaultGroup();
$_SESSION['group'] = $currentGroup;
}

if(isset($_SESSION['period'])){
$currentPeriod = $_SESSION['period'];
} else {
$currentPeriod = $teacher->getDefaultPeriod();
$_SESSION['period'] = $currentPeriod;
}

$currentGroupName = $teacher->getNameOfCurrentGroup($currentGroup, $currentPeriod);
$fullPeriod = $teacher->getFullNameOfCurrentPeriod($currentPeriod);

?>

<!doctype html>
<html lang="en">
<head>
   <meta charset="utf-8">
   <meta name="viewport" content="width=device-width, initial-scale=1">
   <title>Журнал куратора</title>
   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-GLhlTQ8iRABdZLl6O3oVMWSktQOp6b7In1Zl3/Jr59b6EGGoI1aFkw7cmDA6j6gD" crossorigin="anonymous">
   <!-- <link rel="stylesheet" href="img/fontello-c2c4b15f/css/fontello.css">-->
   <link rel="stylesheet" href="../app/web/img/696408cb/css/fontello.css">
   <link rel="stylesheet" href="../app/web/css/xcal.css">
   <link rel="stylesheet" href="../app/web/css/style.css?v=1.31">
   <script src="../app/web/js/jquery-3.6.3.min.js"></script>
</head>
<body class="d-flex flex-column min-vh-100">
<div id="preloader" ></div>
<nav class="navbar navbar-expand-lg bg-body-tertiary">
   <div class="container">

      <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
         <span class="navbar-toggler-icon"></span>
      </button>
      <div class="collapse navbar-collapse" id="navbarSupportedContent">
```

```
        <ul class="navbar-nav me-auto mb-2 mb-lg-0">
            <!-- <li class="nav-item">
                <span id="teacher" data-teacher-id = '<?php  echo $teacher->getTeacherId(); ?>'><?php
                    echo $teacher->getTeacherFIO();
                    ?></span>
            </li> -->
            <li class="nav-item">
                <select id="select_group">
                    <option disabled>Виберіть групу</option>
                    <?php
                    $arr_results = $teacher->getListOfTeacherGroups($currentPeriod);
                    for($i = 0; $i < count($arr_results); $i++){
                    ?>
                    <option value="<?php echo $arr_results[$i]['group_id']; ?>" <?php if($arr_results[$i]['group_id'] == $currentGroup){ echo 'selected';} ?> ><?php echo $arr_results[$i]['group_name']; ?></option>
                    <?php
                    }
                    ?>
                </select>
            </li>
            <li class="nav-item">
                <select  id="select_period">
                    <option disabled>Виберіть період</option>
                    <?php
                      $arr_results = $teacher->getListOfPeriods();
                    for($i = 0; $i < count($arr_results); $i++){
                    ?>
                    <option <?php if($arr_results[$i]["period_id"] == $currentPeriod){ echo 'selected';} ?> value='<?php echo $arr_results[$i]["period_id"]; ?>'><?php echo $arr_results[$i]["period"]; ?></option>
                    <?php
                      }
                    ?>
                </select>
            </li>
            <li>
                <button class="btn btn-outline-primary pdf-print" type="">Word</button>
            </li>
            <li>
                <button class="btn btn-outline-success excel-export" type="">Excel</button>
            </li>
        </ul>

    </div>
    <div id="b12"></div>
    <span id="teacher" data-teacher-id = '<?php  echo $teacher->getTeacherId(); ?>'><?php
                echo $teacherFIO;
                ?></span>
    <form action="user/logout/">
        <button type="submit" class="btn btn-outline-danger logout" type="">Вийти</button>
    </form>
  </div>
</nav>
<div class="container">
  <div class="row">
    <nav>
        <div class="nav nav-tabs" id="nav-tab" role="tablist">
            <button class="nav-link active" id="nav-home-1" data-bs-toggle="tab" data-bs-target="#nav-1" type="button" role="tab" aria-controls="nav-home" aria-selected="true">Загальна інформація</button>
            <button class="nav-link" id="nav-home-2" data-bs-toggle="tab" data-bs-target="#nav-2" type="button" role="tab" aria-controls="nav-profile"
```

```
aria-selected="false">ПЛАН виховної роботи</button>
        <button class="nav-link" id="nav-home-3" data-bs-toggle="tab" data-bs-target="#nav-3" type="button" role="tab" aria-controls="nav-contact" aria-selected="false">ЗВІТ з виховної роботи</button>
        <button class="nav-link" id="nav-home-4" data-bs-toggle="tab" data-bs-target="#nav-4" type="button" role="tab" aria-controls="nav-disabled" aria-selected="false">ОБЛІК присутності</button>
        <button class="nav-link" id="nav-home-5" data-bs-toggle="tab" data-bs-target="#nav-5" type="button" role="tab" aria-controls="nav-disabled" aria-selected="false">АКТИВНІСТЬ студентів</button>
        <button class="nav-link" id="nav-home-6" data-bs-toggle="tab" data-bs-target="#nav-6" type="button" role="tab" aria-controls="nav-disabled" aria-selected="false">ОСОБИСТІ якості</button>
        <button class="nav-link" id="nav-home-7" data-bs-toggle="tab" data-bs-target="#nav-7" type="button" role="tab" aria-controls="nav-disabled" aria-selected="false">ІНДИВІДУАЛЬНА робота</button>
        <button class="nav-link" id="nav-home-8" data-bs-toggle="tab" data-bs-target="#nav-8" type="button" role="tab" aria-controls="nav-disabled" aria-selected="false">РЕЙТИНГОВЕ оцінювання</button>
      </div>
    </nav>
    <div class="tab-content" id="nav-tabContent">
      <div class="tab-pane fade show active" id="nav-1" role="tabpanel" aria-labelledby="nav-home-tab" tabindex="0">
        <h2>ЗАГАЛЬНА ІНФОРМАЦІЯ ПРО СТУДЕНТІВ</h2>
        <table id="tbl1">
          <tr>
            <th rowspan="2" style="width: 26px;">№</th>
            <th rowspan="2" style="width: 230px;">Прізвище, ім’я, по батькові</th>
            <th rowspan="2" style="width: 10px;">Дата народження</th>
            <th rowspan="2">Домашня адреса</th>
            <th rowspan="2">Місце проживання під час навчання</th>
            <th colspan="2">Додаткова інформація про студента</th>
            <th colspan="2">Інформація про батька</th>
            <th colspan="2">Інформація про матір</th>
          </tr>
          <tr>
            <th>Електронна пошта</th>
            <th>Номер телефону</th>
            <th>Прізвище, Ім’я, по батькові</th>
            <th>Контактний телефон</th>
            <th>Прізвище, Ім’я, по батькові</th>
            <th>Контактний телефон</th>
          </tr>
          <?php
          $t1arr = (object)["group"=>$currentGroup, "period"=>$currentPeriod ];
          $table01 = new \app\Tables\Table01($t1arr);
          $row_count = 0;
          $results1 = $table01->getAllData();
          for($i = 0; $i < count($results1); $i++ ){
          $row_count++;
          ?>
          <tr>
            <td data-editable="no"><?php echo $row_count; ?></td>
            <td data-editable="no" data-element-id='<?php echo $results1[$i]["tbl01_id"]; ?>'><?php echo $results1[$i]["student_fio"]; ?></td>
            <?php
              $source = $results1[$i]["birth_date"];
```

```
                $date = new DateTime($source);
                ?>
            <td data-editable="no"><?php echo $date->format('d.m.Y'); ?></td>
            <td data-editable="yes"><?php echo $results1[$i]["home_address"]; ?></td>
            <td data-editable="yes"><?php echo $results1[$i]["living_address"]; ?></td>
            <td data-editable="yes"><?php echo $results1[$i]["student_email"]; ?></td>
            <td data-editable="yes"><?php echo $results1[$i]["student_tel"]; ?></td>
            <td data-editable="yes"><?php echo $results1[$i]["father_fio"]; ?></td>
            <td data-editable="yes"><?php echo $results1[$i]["father_tel"]; ?></td>
            <td data-editable="yes"><?php echo $results1[$i]["mother_fio"]; ?></td>
            <td data-editable="yes"><?php echo $results1[$i]["mather_tel"]; ?></td>
          </tr>
          <?php
          }
          ?>
        </table>
        <div id="a12"></div>

      </div>
      <div class="tab-pane fade" id="nav-2" role="tabpanel" aria-labelledby="nav-profile-tab" tabindex="0">
        <h2>ПЛАН виховної роботи в <?php echo $currentGroupName; ?> групі на <?php echo $fullPeriod; ?> навчальний рік</h2>
        <table id="tbl2">
          <tr>
            <th style="width: 26px;">№</th>
            <th>Назва заходу</th>
            <th style="width: 10px;">Термін виконання <i id="data2" class="demo-icon icon-sorting-1" data-toggle="tooltip" data-placement="top" title="сортувати за зростанням"></i></th>
            <th>Відповідальні за виконання</th>
            <th style="width: 45px;">Перенести в звіт</th>
            <th style="width: 45px;"></th>
          </tr>
          <?php
            $t2arr = (object)["group"=>$currentGroup, "period"=>$currentPeriod ];
            $table2 = new \app\Tables\Table02($t2arr);
            $results2 = $table2->getAllData();
            $row_count = 1;

            for($i = 0; $i < count($results2); $i++){
            ?>
            <tr>
              <td data-editable="no"><?php echo $row_count; ?></td>
              <td data-editable="yes" data-element-id='<?php echo $results2[$i]["tbl02_id"]; ?>'><?php echo $results2[$i]["event"]; ?></td>
              <td data-editable="yes" date-type="date" ><?php
                if($results2[$i]["event_date"] == '2000-01-01'){
                  echo "";
                } else {
                  echo date('d.m.Y', strtotime(str_replace('.', '-', $results2[$i]["event_date"])));
                }
                ?></td>
              <td data-editable="yes"><?php echo $results2[$i]["in_charge"]; ?></td>
              <td data-editable="no"><input type="checkbox" class="to-report"
                <?php
                  if($results2[$i]["to_report"]){
                    echo 'checked';
                  } ?>>
              </td>
```

```
                <td data-editable="no" data-type-element="delete"><i class="demo-icon icon-cancel-1"></i></td>
            </tr>
            <?php
              $row_count++;
              }
            ?>
            <tr>
              <td data-editable="no"><?php echo $row_count; ?></td>
              <td data-editable="yes" data-tmp-id='<?php echo microtime(true).rand(1000000, 100000000); ?>'></td>
              <td data-editable="yes" date-type="date"></td>
              <td data-editable="yes"></td>
              <td data-editable="no"><input type="checkbox" class="to-report" disabled></td>
              <td data-editable="no"><i class="demo-icon icon-cancel-1 icon-cancel-inactive"></i></td>
            </tr>
          </table>

        </div>
        <div class="tab-pane fade" id="nav-3" role="tabpanel" aria-labelledby="nav-contact-tab" tabindex="0">
          <h2>ЗВІТ з виховної роботи в <?php echo $currentGroupName; ?> групі за <?php echo $fullPeriod; ?> навчальний рік</h2>
          <?php
          $t3arr = (object)["group"=>$currentGroup, "period"=>$currentPeriod ];
          $table3 = new \app\Tables\Table03($t3arr);
          ?>
          <table id="tbl3" data-method='<?php echo $table3->getMethod(); ?>'>
            <tr>
              <th style="width: 26px;">№</th>
              <th style="width: 10px;"></th>
              <th>Назва заходу</th>
              <th style="width: 10px;">Термін виконання <i id="data3" class="demo-icon icon-sorting-1" data-toggle="tooltip" data-placement="top" title="сортувати за зростанням"></i></th>
              <th>Відповідальні за виконання</th>
              <th style="width: 120px;">Відвідування</th>
              <th style="width: 45px;"></th>
            </tr>
            <?php

            $results3 = $table3->getAllData();
            $row_count = 1;

            for($i = 0; $i < count($results3); $i++){
            ?>
            <tr>
              <td data-editable="no"><?php echo $row_count; ?></td>
              <?php
                if($results3[$i]["standard_event_id"] > 0){
              ?>
              <td data-editable="no" ><input class="standard-event" type="checkbox" disabled checked></td>
              <td data-editable="no" data-element-id='<?php echo $results3[$i]["tbl03_id"]; ?>' data-event-id='<?php echo $results3[$i]["standard_event_id"]; ?>'><?php echo $results3[$i]["event"]; ?></td>
              <?php
                } else {
              ?>
              <td data-editable="no" ><input class="standard-event" type="checkbox"></td>
              <td data-editable="yes" data-element-id='<?php echo $results3[$i]["tbl03_id"]; ?>' data-event-id='<?php echo $results3[$i]["standard_event_id"]; ?>'><?php echo $results3[$i]["event"]; ?></td>
              <?php
```

```
                }
            ?>
            <td data-editable="yes" date-type="date"><?php
                if($results3[$i]["event_date"] == '2000-01-01'){
                    echo "";
                } else {
                    echo date('d.m.Y', strtotime(str_replace('.', '-', $results3[$i]["event_date"])));
                    //echo $arr_results[$i]["event_date"];
                }
                ?></td>
            <td data-editable="yes"><?php echo $results3[$i]["in_charge"]; ?></td>
            <td data-editable="no"><button class="btn btn-outline-success event-visit" type="button" data-bs-toggle="modal" data-bs-target="#staticBackdrop">Відвідування</button></td>
            <td data-editable="no" data-type-element="delete"><i class="demo-icon icon-cancel-1"></i></td>
        </tr>
        <?php
        $row_count++;
    }
    ?>
        <tr>
            <td data-editable="no"><?php echo $row_count; ?></td>
            <td data-editable="no" ><input type="checkbox" class="standard-event"></td>
            <td data-editable="yes" data-tmp-id='<?php echo microtime(true).rand(1000000, 100000000); ?>' data-event-id='0'></td>
            <td data-editable="yes" date-type="date"></td>
            <td data-editable="yes"></td>
            <td data-editable="no"><button class="btn btn-outline-success event-visit" type="button" data-bs-toggle="modal" >Відвідування</button></td> <!-- data-bs-target="#staticBackdrop" -->
            <td data-editable="no"><i class="demo-icon icon-cancel-1 icon-cancel-inactive"></i></td>
        </tr>
    </table>
    <div id="forTable3Data"></div>


    <!-- Modal -->
    <div class="modal fade" id="staticBackdrop" data-bs-backdrop="static" data-bs-keyboard="false" tabindex="-1" aria-labelledby="staticBackdropLabel" aria-hidden="true">
        <div class="modal-dialog">
            <div class="modal-content">
                <div class="modal-header">
                    <h1 class="modal-title fs-5" id="staticBackdropLabel">Група <?php echo $currentGroupName; ?></h1>
                    <button type="button" class="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>
                </div>
                <div class="modal-body">
                    <div id="forGroupListData"></div>
                </div>
                <div class="modal-footer">
                    <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Закрити</button>
                </div>
            </div>
        </div>
    </div>
```

```
<div class="tab-pane fade" id="nav-4" role="tabpanel" aria-labelledby="nav-disabled-tab" tabindex="0">
<h2>ОБЛІК ПРИСУТНОСТІ СТУДЕНТІВ НА ВИХОВНИХ ЗАХОДАХ У <?php echo $fullPeriod; ?> НАВЧАЛЬНОМУ РОЦІ</h2>
<div id="forTable04"></div>
</div>
<div class="tab-pane fade" id="nav-5" role="tabpanel" aria-labelledby="nav-disabled-tab" tabindex="0">
<h2>ГРОМАДСЬКА АКТИВНІСТЬ СТУДЕНТІВ ГРУПИ</h2>
<table id="tbl5">
<tr>
<th>№</th>
<th>Прізвище, ім’я та по батькові</th>
<th>Роль, функції та обов’язки студента в групі</th>
<th>Робота в клубах за інтересами, гуртках, спортивних секціях</th>
<th>Участь у діяльності громадських організацій</th>
</tr>
<?php
$t5arr = (object)["group"=>$currentGroup, "period"=>$currentPeriod ];
$table05 = new \app\Tables\Table05($t5arr);
$table05->checkTable5ForActualData();
$row_count = 0;
$results5 = $table05->getAllData();
for($i = 0; $i < count($results5); $i++ ){
$row_count++;
?>
<tr>
<td data-editable="no"><?php echo $row_count; ?></td>
<td data-editable="no" data-element-id='<?php echo $results5[$i]['tbl05_id']; ?>'><?php echo $results5[$i]['student_fio']; ?></td>
<td data-editable="yes"><?php echo $results5[$i]['stud_functions']; ?></td>
<td data-editable="yes"><?php echo $results5[$i]['stud_interests']; ?></td>
<td data-editable="yes"><?php echo $results5[$i]['public_organizations']; ?></td>
</tr>
<?php
}
?>
</table>

</div>
<div class="tab-pane fade" id="nav-6" role="tabpanel" aria-labelledby="nav-disabled-tab" tabindex="0">
<h2>ОСОБИСТІ ЯКОСТІ СТУДЕНТА</h2>
<table id="tbl6">
<tr>
<th style="width: 26px;" >№</th>
<th style="width: 390px;">Прізвище, ім’я та по батькові</th>
<th style="width: 30px;"></th>
<th >Риси особистості</th>
<th style="width: 107px;"></th>
<th style="width: 130px;">Рівень вихованості</th>
</tr>
<?php
$t6arr = (object)["group"=>$currentGroup, "period"=>$currentPeriod ];
$table06 = new \app\Tables\Table06($t6arr);
$table06->checkTable6ForActualData();
$traitList = $table06->getTraitList();
$row_count = 0;
$results6 = $table06->getAllData();
for($i = 0; $i < count($results6); $i++ ){
```

```
            $arr = (array)json_decode($results6[$i]['positive_traits']);

            $row_count++;
            ?>
            <tr>
              <td data-editable="no"><?php echo $row_count; ?></td>
              <td data-editable="no" data-element-id='<?php echo $results6[$i]['tbl06_id']; ?>'><?php echo $results6[$i]['student_fio']; ?></td>
              <td data-editable="no"><button class="btn btn-outline-success traits-list" type="button" data-bs-toggle="modal" data-bs-target="#staticBackdrop2">Риси</button></td>
              <td data-editable="no"><?php echo $table06->traitParsing($traitList, $arr);  ?></td>
              <td data-editable="no"><button class="btn btn-outline-secondary traits-clear" type="button">Очистити все</button></td>
              <td data-editable="no">
                <select class="education-level">
                  <?php $levelValue = $results6[$i]['education_level']; ?>
                  <option value="5" <?php if($levelValue == 5 || $levelValue < 3 || $levelValue > 5) echo "selected"; ?> >високий</option>
                  <option value="4" <?php if($levelValue == 4) echo "selected"; ?> >середній</option>
                  <option value="3" <?php if($levelValue == 3) echo "selected"; ?> >низький</option>
                </select>
              </td>
            </tr>
            <?php
            }
            ?>
          </table>
          <div id="forTable06"></div>
          <!-- Modal -->
          <div class="modal fade" id="staticBackdrop2" data-bs-backdrop="static" data-bs-keyboard="false" tabindex="-1" aria-labelledby="staticBackdropLabel2" aria-hidden="true">
            <div class="modal-dialog modal-lg">
              <div class="modal-content">
                <div class="modal-header">
                  <h6 class="modal-title" id="staticBackdropLabel2" style="font-size: 16px; color: lightseagreen;">Виберіть 1 характеристику для кожної категорії</h6>
                  <button type="button" class="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>
                </div>
                <div class="modal-body">
                  <div class="preloader"></div>
                  <div id="forTraitsData"></div>
                </div>
                <div class="modal-footer">
                  <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Закрити</button>
                </div>
              </div>
            </div>
          </div>

        </div>
        <div class="tab-pane fade" id="nav-7" role="tabpanel" aria-labelledby="nav-disabled-tab" tabindex="0">
          <h2>ПРОВЕДЕННЯ ІНДИВІДУАЛЬНОЇ РОБОТИ ЗІ СТУДЕНТАМИ ГРУПИ</h2>
          <?php
          $t7arr = (object)["group"=>$currentGroup, "period"=>$currentPeriod ];
          $table07 = new \app\Tables\Table07($t7arr);
          $listOfStudents1 = $table07->getListOfStudentsInGroup();
```

```
$numberOfStudents = count($listOfStudents1);
$str7 = "<select class='list-of-students'> <option disabled selected>Виберіть студента</option>";
for($jj = 0; $jj < $numberOfStudents; $jj++){
$str7 .= "<option value='" . $listOfStudents1[$jj]['student_id'] . "' >" . $listOfStudents1[$jj]['student_fio'] . "</option>";
}
$str7 .= "</select>";

?>

<table id="tbl7">
<tr>
<th style="width: 26px;">№</th>
<th style="width: 10px;">Дата проведення</th>
<th style="width: 350px;">Прізвище, ім’я та по батькові</th>
<th>Зміст індивідуальної роботи</th>
<th style="width: 80px;">Кількість годин</th>
<th style="width: 45px;"></th>
</tr>
<?php
$row_count = 0;
$results7 = $table07->getAllData();
for($i = 0; $i < count($results7); $i++ ){
$row_count++;
?>
<tr>
<td data-editable="no"><?php echo $row_count; ?></td>
<td data-editable="yes" date-type="date" data-element-id='<?php echo $results7[$i]["tbl07_id"]; ?>'><?php
if($results7[$i]["event_date"] == '2000-01-01'){
echo "";
} else {
echo date('d.m.Y', strtotime(str_replace('.', '-', $results7[$i]["event_date"])));
}
?></td>
<td data-editable="no" student-id='<?php echo $results7[$i]["student_id"]; ?>'>
<?php echo $str7; ?>
</td>
<td data-editable="yes"><?php echo $results7[$i]["individual_work"]; ?></td>
<td data-editable="yes"><?php echo $results7[$i]["hours"]; ?></td>
<td data-editable="no"><i class="demo-icon icon-cancel-1"></i></td>
</tr>
<?php
}
?>
<tr>
<td data-editable="no"><?php echo ++$row_count; ?></td>
<td data-editable="yes" date-type="date" data-tmp-id='<?php echo microtime(true).rand(1000000, 100000000); ?>'></td>
<td data-editable="no">
<?php echo $str7; ?>
</td>
<td data-editable="yes"></td>
<td data-editable="yes"></td>
<td data-editable="no"><i class="demo-icon icon-cancel-1 icon-cancel-inactive"></i></td>
</tr>
</table>
</div>
<div class="tab-pane fade" id="nav-8" role="tabpanel" aria-labelledby="nav-disabled-tab" tabindex="0">
<?php
$t8arr = (object)["group"=>$currentGroup, "period"=>$currentPeriod ];
```

```
$table08 = new \app\Tables\Table08($t8arr);
$results8 = $table08->getAllData();
?>
<h2>РЕЗУЛЬТАТИ РЕЙТИНГОВОГО ОЦІНЮВАННЯ СТУДЕНТІВ ГРУПИ <?php echo $currentGroupName; ?> ЗА <?php echo $fullPeriod; ?> РІК</h2>
<table id="tbl8">
  <tr>
    <th style="width: 26px;">№</th>
    <th>Прізвище, ім’я та по батькові <i id="fio" class="demo-icon icon-sorting-1" data-toggle="tooltip" data-placement="top" title="сортувати за алфавітом"></i></th>
    <th>Наукова робота</th>
    <th>Громадська робота</th>
    <th>Спортивна діяльність</th>
    <th>Культурно-масова діяльність</th>
    <th>Соціальна активність (пасивність)</th>
    <th>Загальний результат <i id="total" class="demo-icon icon-sorting-1" data-toggle="tooltip" data-placement="top" title="сортувати за спаданням"></i></th>
  </tr>
  <?php
    $row_count = 1;
    for($k = 0; $k < count($results8); $k++){
  ?>
  <tr>
    <td data-editable="no"><?php echo $row_count; ?></td>
    <td data-editable="no" data-element-id='<?php echo $results8[$k]["tbl08_id"]; ?>'><?php echo $results8[$k]['student_fio']; ?></td>
    <td data-editable="no" data-select-value='<?php echo $results8[$k]['scientific_work']; ?>'>
      <select class="student-evaluation">
        <option value="5">5</option>
        <option value="4">4</option>
        <option value="3">3</option>
        <option value="2">2</option>
        <option value="1">1</option>
        <option value="0">0</option>
      </select>
    </td>
    <td data-editable="no" data-select-value='<?php echo $results8[$k]['public_work']; ?>'>
      <select class="student-evaluation">
        <option value="5">5</option>
        <option value="4">4</option>
        <option value="3">3</option>
        <option value="2">2</option>
        <option value="1">1</option>
        <option value="0">0</option>
      </select>
    </td>
    <td data-editable="no" data-select-value='<?php echo $results8[$k]['sports_activities']; ?>'>
      <select class="student-evaluation">
        <option value="5">5</option>
        <option value="4">4</option>
        <option value="3">3</option>
        <option value="2">2</option>
        <option value="1">1</option>
        <option value="0">0</option>
      </select>
    </td>
    <td data-editable="no" data-select-value='<?php echo $results8[$k]['cultural_mass_activity']; ?>'>
      <select class="student-evaluation">
        <option value="5">5</option>
        <option value="4">4</option>
        <option value="3">3</option>
        <option value="2">2</option>
        <option value="1">1</option>
        <option value="0">0</option>
      </select>
    </td>
```

```
                <td data-editable="no" data-select-value='<?php echo $results8[$k]['social_activity']; ?>'>
                <select class="student-evaluation">
                  <option value="5">5</option>
                  <option value="4">4</option>
                  <option value="3">3</option>
                  <option value="2">2</option>
                  <option value="1">1</option>
                  <option value="0">0</option>
                </select>
                </td>
                <td data-editable="no" data-select-value='<?php echo $results8[$k]['total_results']; ?>'>
                <select class="student-evaluation">
                  <option value="5">5</option>
                  <option value="4">4</option>
                  <option value="3">3</option>
                  <option value="2">2</option>
                  <option value="1">1</option>
                  <option value="0">0</option>
                </select>
                </td>
              </tr>
              <?php
                  $row_count++;
                }
              ?>
            </table>
          </div>
        </div>
      </div>

<footer class="py-3 my-4 mt-auto">
  <ul class="nav justify-content-center border-bottom pb-3 mb-3">
    <li class="nav-item"><a href="https://help.udau.edu.ua/index.php?a=add&category=1" target="_blank" class="nav-link px-2 text-muted">Запит на допомогу в центр підтримки УНУС</a></li>
  </ul>
  <p class="text-center text-muted">Уманський НУС <?php echo date('Y'); ?></p>
</footer>

<!-- Modal -->
<div class="modal fade" id="eraseRecord" tabindex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <!-- <h1 class="modal-title fs-5" id="exampleModalLabel"></h1>-->
        <button type="button" class="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>
      </div>
      <div class="modal-body mx-auto">
        <p style="text-align: center; font-size: 20px;">Запис буде <span style="color: #dc3545; font-weight: 700;" >безповоротно</span> видалено!</p>
        <!-- <p style="text-align: center; font-size: 16px;">Також буде видалено зв'язаний з ним запис у звіті з виховної роботи, якщо такий є!</p>-->
        <button type="button" id="deleteRecordTbl1" class="btn btn-outline-danger" data-bs-dismiss="modal" style="margin: auto; display: block; margin-top: 30px; margin-bottom: 20px;">Видалити запис</button>
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Close</button>
      </div>
    </div>
  </div>
</div>
<!-- Modal -->
<div class="modal fade" id="inform" data-bs-backdrop="static" data-bs-keyboard="false" tabindex="-1" aria-labelledby="staticBackdropLabel" aria-hidden="true">
```

```
  <div class="modal-dialog modal-dialog-centered">
    <div class="modal-content">
      <div class="modal-header">
        <h1 class="modal-title fs-5 align-content-center" id="staticBackdropLabel" ></h1>
        <button type="button" class="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>
      </div>
      <div class="modal-body" style="text-align: center; font-size: 18px;">
        <span style="color: #DC3545;">Помилка з'єднання з сервером!</span><br>
        <span style="color: #495057">Останні зміни не збережені.</span><br>
        <span style="color: #495057">Спробуйте пізніше.</span><br>
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Close</button>
      </div>
    </div>
  </div>
</div>
```

```
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js" integrity="sha384-w76AqPfDkMBDXo30jS1Sgez6pr3x5MlQ1ZAGC+nuZB+EYdgRZgiwxhTBTkF7CXvN" crossorigin="anonymous"></script>
<script src="../app/web/js/xcal-en.js?v=1.30"></script>
<script defer src="../app/web/js/main.js?v=1.32"></script>
<script defer src="../app/web/js/table02.js?v=1.30"></script>
<script defer src="../app/web/js/table03.js?v=1.31"></script>
<script defer src="../app/web/js/table04.js?v=1.30"></script>
<script defer src="../app/web/js/table06.js?v=1.33"></script>
<script defer src="../app/web/js/table07.js?v=1.30"></script>
<script defer src="../app/web/js/table08.js?v=1.30"></script>
<script defer src="../app/web/js/excel.js?v=1.30"></script>
</body>
</html>
```

ДОДАТОК Б

ЛІСТИНГ МОДУЛЯ ІНТЕГРАЦІЇ ІЗ АСУ

```
<?php

namespace app;

use app\DB\DB;
use app\models\StudentGroup;
use app\tables\Table05;
use app\tables\Table06;
use app\Tables\Table08;

class ASU extends StudentGroup
{

    protected $mkr = null;
    protected $logFile = "";
    protected $fileRegister=[];
    //protected $user = null;
    protected $adminU = "test ";
    protected $adminP = "test";
    protected $adminList = [1407,7];

    public $config = [];
    public $plh = [];
    public $table = [];

    public function getError()
    {
        return $_SESSION['error'];
    }

    protected function getToken(){
        return $this->mkr->getToken();
    }

    protected function getUserInfo()
    {
        return $this->mkr->getUserInfo();
    }

    protected function checkIfAllCuratorsFromASUinDB()
    {
//      Array
//      (
//          [0] => stdClass Object
//              (
//                  [kgrp1] => 290
//                  [kgrp2] => 1272
//                  )
//
//          [1] => stdClass Object
//              (
//                  [kgrp1] => 562
//                  [kgrp2] => 1284
//                  )
//
//          [2] => stdClass Object
//              (
//                  [kgrp1] => 58
//                  [kgrp2] => 1329
//                  )
//
//          [3] => stdClass Object
//              (
//                  [kgrp1] => 221
//                  [kgrp2] => 1285
//                  )

        $curatorsArr = $this->mkr->getListCurators();
        $info = json_decode($curatorsArr['response']);
        $info = serialize($info);
        //в info:
        //a:211:{i:0;O:8:"stdClass":2:{s:5:"kgrp1";i:290;s:5:"kgrp2";i:1272;}i:1;O:8:"stdClass":2:{s:5:"kgrp1";i:562;s:5:"kgrp2";i:1284;}i:2;O:8:"stdClass":2:{s:5:"kgrp1";i:58;s:5:"kgrp2";i:1329;}i:3;O:8:"stdClass":2:{s:5:"kgrp1";i:221;s:5:"kgrp2";i:1285;}
```

```
        $sql = "SELECT * FROM `curator_group` WHERE 1";
    $results = DB::getInstance()->getfromBD($sql);
    $results = serialize($results);

    //"kgrp1";i:97;
    $pattern0 = '/"kgrp1";i:\d+;/';
    preg_match_all($pattern0, $info, $matches0);

    $asuIdArr = [];
    foreach ($matches0[0] as $teacher){
      preg_match_all('/\d+/', $teacher, $tmp_asu_id);
      array_push($asuIdArr, $tmp_asu_id[0][1]);
    }

    asort($asuIdArr);
     $curatorIdActualList = array_unique($asuIdArr); // список id кураторов из АСУ

    ////////////////////////////////////////////////////////////////////////////////
        // надо в таблице curator_group выявить кураторов у которых раньше были группы, но на текущий момент у них групп не осталось
    // Таких кураторов надо удалить
      $sql00 = "SELECT * FROM `curator_group` WHERE 1";
                $results00 = DB::getInstance()->getfromBD($sql00);
    $dataArr = [];
    foreach ($results00 as $data){
      array_push($dataArr, $data['curator_asu_id']);
    }
    asort($dataArr);
       $curatorIdInDB = array_unique($dataArr); // список id кураторов в БД

      foreach ($curatorIdInDB as $DBcuratorId){ // выявление в БД кураторов у которых не осталось групп
                                    if(!in_array($DBcuratorId, $curatorIdActualList)){
//                  $sql_000 = "DELETE FROM `curator_group` WHERE `curator_asu_id` = " . $DBcuratorId;
//                                    DB::getInstance()->executeQuery($sql_000);

        }
      }

    foreach ($curatorIdActualList as $tmp_id){
            $pattern1 = '/"kgrp1";i:' . $tmp_id . ';s:5:"kgrp2";i:\d+;/';
      preg_match_all($pattern1, $info, $matches1);

      $groups1 = [];
      foreach ($matches1[0] as $teacher){
                  preg_match_all('/\d+/', $teacher, $tmp_group);
          array_push($groups1, $tmp_group[0][4]); // в $groups1 собрали все группы текущего куратора из АСУ
      }

      $pattern2 = '/"curator_asu_id";s:\d+:"' . $tmp_id . '";s:12:"group_asu_id";s:\d+:"\d+";/';
                  preg_match_all($pattern2, $results, $matches2);

      $groups2 = [];
      foreach ($matches2[0] as $teacher){
                  preg_match_all('/\d+/', $teacher, $tmp_group);
          array_push($groups2, $tmp_group[0][4]); // в $groups2 собрали все группы текущего куратора из БД
      }

      foreach ($groups1 as $currentGroupASU){
        if(!in_array($currentGroupASU, $groups2)){
```

```
                    $outOfStudyProcess = $this->checkIfGroupIsNotInStudyProcess($currentGroupASU);
                    $sql10 = "INSERT INTO `curator_group`(`order_id`, `curator_asu_id`, `group_asu_id`, `teacher_id`, `outOfStudyProcess`, `global_group_name`, `group_name`)
                    VALUES (NULL ,'" . $tmp_id . "','" . $currentGroupASU . "', NULL, " . $outOfStudyProcess . ", NULL, NULL)";
                    DB::getInstance()->executeQuery($sql10);
                }
            }// проверяем все ли группы из АСУ для текущего куратора в БД, если нет - добавляем в БД

            foreach ($groups2 as $currentGroupBD){// уточнить работу ЭТОГО цикла !!! слишком много запросов в БД!!!
                if(!in_array($currentGroupBD, $groups1)){
                    $sql21 = "SELECT * FROM `curator_group` WHERE `curator_asu_id`=" . $tmp_id . " AND `group_asu_id`=" . $currentGroupBD;
                    $results21 = DB::getInstance()->getfromBD($sql21);
                    $order_id = $results21[0]['order_id'];
                    if($order_id){
                        $sql22 = "DELETE FROM `curator_group` WHERE `order_id`=" . $order_id;
                        DB::getInstance()->executeQuery($sql22);

                        $sql23 = "SELECT * FROM `groups` WHERE `group_asu_id`=" . $currentGroupBD;
                        $results23 = DB::getInstance()->getfromBD($sql23);
                        if($results23){
                            $sql24 = "DELETE FROM `teachers_groups` WHERE `group_id`=" . $results23[0]['group_id'];
                            DB::getInstance()->executeQuery($sql24);
                        }
                    }
                }
            } // проверяем лишние группы в БД для текущего куратора, если такой группы нет в списке групп из АСУ, то такую группу удалить из БД

        } // end FOREACH()
        // по окончанию работы цикла в БД получаем актуальный список групп для каждого куратора:
        // если куратор новый - в БД добавляется он и его группы
        // если у куратора изменился состав групп, но их у него остается больше одной - в БД обновляется до актуального
        // если у куратора стало 0 групп, то он остается в БД

        ////////////////////////////////////////
        ///
        /// к текущему моменту в таблице curator_group содержится актуальный список групп для каждого куратора как это есть в АСУ
            /// надо синхронизировать таблицу teachers_groups с таблицей curator_group
            /// в таблице teachers_groups отображаются группы кураторов, которые хотя бы раз авторизовались в журнале
        /// в таблице teachers отображаются кураторы, которые хотя бы раз зашли в журнал

        $sql40 = "SELECT * FROM `teachers` WHERE 1";
            $results40 = DB::getInstance()->getfromBD($sql40);

        foreach ($results40 as $teacher){
            $teacher_id = $teacher['teacher_id']; // id куратора в БД
            $teacher_asu_id = $teacher['teacher_asu_id']; // id из АСУ
```

```
                $teacher_real_id = $teacher['teacher_real_id']; // другое id из АСУ

                $sql41 = "SELECT * FROM `teachers_groups` WHERE `teacher_id`=" . $teacher_id;
                $results41 = DB::getInstance()->getfromBD($sql41);

        $teacherGroups41 = [];
        if($results41){
          foreach ($results41 as $teacherGroup){
                array_push($teacherGroups41, $teacherGroup['group_id']);
          }
        }// в $teacherGroups41 содержится список групп текущего куратора согласно БД

        $sql42 = "SELECT * FROM `curator_group` WHERE `curator_asu_id`=" . $teacher_asu_id;
                $results42 = DB::getInstance()->getfromBD($sql42);
          //в $results42 содержится актуальная с АСУ информация по группам для текущего куратора
        $teacherGroups42 = [];
        if($results42){
                // на каком этапе таблица groups наполняется данными???
          foreach ($results42 as $curatorGroup){
            $sql42_1 = "SELECT * FROM `groups` WHERE `group_asu_id`=" . $curatorGroup['group_asu_id'];
                $results42_1 = DB::getInstance()->getfromBD($sql42_1); // в $results42_1 содержится дополнительная инфа по актуальным группам для текущего куратора
            if($results42_1){
                array_push($teacherGroups42, $results42_1[0]['group_id']);
            }
          } // в массив $teacherGroups42 складываются id группы (group_id) в БД

        foreach($teacherGroups41 as $groupId){
          if(!in_array($groupId, $teacherGroups42)){
                $sql50 = "DELETE FROM `teachers_groups` WHERE `teacher_id`=" . $teacher_id . " AND `group_id`=" . $groupId;
            DB::getInstance()->executeQuery($sql50);
          }
        }
      }

    }

    protected function isCurator($teacher_asu_id)
    {
        $sql = "SELECT * FROM `curator_group` WHERE `curator_asu_id`=" . $teacher_asu_id;
      $result = DB::getInstance()->getfromBD($sql);
      if($result){
        return 1;
      } else {
        return 0;
      }
    }

                protected function registrateTimeOfVisit($teacher_asu_id)
    {
      //$sql = "UPDATE `teachers` SET `visit_time`= now()  WHERE `teacher_asu_id`=" . $teacher_asu_id;
      //DB::getInstance()->executeQuery($sql);

       $sql = "SELECT * FROM `teachers` WHERE `teacher_asu_id`=" . $teacher_asu_id ;
      $result = DB::getInstance()->getfromBD($sql);

          $currentDate = date('d.m.Y'); // Получить текущую дату в формате "день.месяц.год"
          $currentTime = date('H:i:s'); // Получить текущее время в формате "часы:минуты:секунды"
```

```
        $dateNow = $currentDate . " " . $currentTime;

        if($result){
            $arr = json_decode($result[0]['visit_time']);
            if(is_array($arr)){
                array_push($arr, $dateNow);
            } else {
                $arr = [];
                array_push($arr, $dateNow);
            }
        } else {
            $arr = [];
            array_push($arr, $dateNow);
        }

        if(count($arr) > 10){
            array_shift($arr);
            //array_values($arr);
        }

        $arrJson = json_encode($arr);

        $sql = "UPDATE `teachers` SET `visit_time`= '" .
$arrJson . "' WHERE `teacher_asu_id`=" .
$teacher_asu_id;
        DB::getInstance()->executeQuery($sql);
    }

    protected  function removeParentheses($string)
    {
        $result = preg_replace('/\([^)]+\)/', '', $string);
        return trim($result);
    }

    ////////////////////////////////test//////////////////
                                public function
checkIfAllCuratorsFromASUinDB2Test()                //
AdminController / actionTest01()
    {
        $this->mkr = new MKR();
        $this->mkr->setUser('user1000000971');
        $this->mkr->setPassword('user1000000971');
        $this->getToken();

        $this->checkIfAllCuratorsFromASUinDB();

    }

    ////////////////////////////////test//////////////////
                                public function
checkStudentFromASU2Test($user_id)                //
AdminController / actionTest01()
    {

        $this->mkr = new MKR();
        $this->mkr->setUser('user1000000971');
        $this->mkr->setPassword('user1000000971');
        $this->getToken();

                        $studentInfo = $this->mkr-
>getStudentInfo25($user_id);
        $info = json_decode($studentInfo['response']);
//        $info = serialize($info);
        echo "<pre>";
        print_r($info);
        echo "</pre>";
    }

                                public function
getGroupsForCuratorFromASU2($data)                //
$teacher_asu_id
    {
//        $this->mkr = new MKR();
//        $this->mkr->setUser('user1000000971');
//        $this->mkr->setPassword('user1000000971');
//        $this->getToken();

        $teacher_id = $data['teacher_id'];
        $teacher_asu_id = $data['teacher_asu_id'];

            $sql2 = "SELECT * FROM `curator_group`
WHERE `curator_asu_id`=" . $teacher_asu_id;
        $result2 = DB::getInstance()->getfromBD($sql2);
// получили данные по группам текущего куратора
```

```
for($q = 0; $q < count($result2); $q++){
    $group_asu_id = $result2[$q]['group_asu_id'];

     $sql3 = "SELECT * FROM `groups` WHERE `group_asu_id`=" . $group_asu_id;
                $result3 = DB::getInstance()->getfromBD($sql3); // получили данные по текущей группе: id в БД и стртовое название
    $group_id = 0;
    $studentsInGroup = null;

    if($result3){
        $group_id = $result3[0]['group_id'];
                $globalName = $result3[0]['group_start_name'];
    } else {
                $studentsInGroup = $this->mkr->getStudentsInGroup2($group_asu_id);
                $studentsInGroup = json_decode($studentsInGroup['response']);
                $globalName = $studentsInGroup[0]->globalName;

                $sql31 = "INSERT INTO `groups`(`group_id`, `group_asu_id`, `group_start_name`, `faculty`) VALUES (NULL,'" . $group_asu_id . "','" . $globalName . "',NULL)";
        DB::getInstance()->executeQuery($sql31);

            $sql32 = "SELECT * FROM `groups` WHERE `group_asu_id`=" . $group_asu_id;
                $result32 = DB::getInstance()->getfromBD($sql32);
        $group_id = $result32[0]['group_id'];
    }

                $sql45 = "SELECT * FROM `teachers_groups` WHERE `teacher_id`='" . $teacher_id . "' AND `group_id`='" . $group_id . "'";
                $result45 = DB::getInstance()->getfromBD($sql45);
        if (!$result45) {
                    $sql45 = "INSERT INTO `teachers_groups`(`teacher_id`, `group_id`, `tg_id`) VALUES ('" . $teacher_id . "','" . $group_id . "',NULL)";
            DB::getInstance()->executeQuery($sql45);
        }

        ////////////////////////////////////////////////////////
                $studentsArr = $this->mkr->getStudentsInGroup5($group_asu_id);
                $students = json_decode($studentsArr['response']);
        try {
            if($students){
                for ($ii = 0; $ii < count($students); $ii++){
                $currentId = $students[$ii]->id;
                $realId = $students[$ii]->realid;
                    $studentFIO = $students[$ii]->ua->lastName . " " . $students[$ii]->ua->firstName . " " . $students[$ii]->ua->secondName;
                    $studBirthArr = explode('.', $students[$ii]->birthday);
                $studentBirthday = $studBirthArr[2] . "-" . $studBirthArr[1] . "-" . $studBirthArr[0];

                    $sql52_1 = "SELECT * FROM `students` WHERE `current_id`= " . $currentId . " AND `group_id`=" . $group_id . " AND `period_id`=" . $this->getCurrentPeriod();
                    $result52_1 = DB::getInstance()->getfromBD($sql52_1);

                if(!$result52_1){
                        $sql53 = "INSERT INTO `students`(`student_id`, `asu_id`, `current_id`, `student_fio`, `birth_date`, `group_id`, `period_id`) VALUES (NULL ,'" . $realId . "','" . $currentId . "','" . $studentFIO . "','" . $studentBirthday . "','" . $group_id . "','" . $this->getCurrentPeriod() . "')";
```

```
                                DB::getInstance()->executeQuery($sql53);
                    }
                }
            } else {

            }

        } catch (Exception $e){
            echo 'Error: ',  $e->getMessage(), "\n";
        }

        $sql4 = "SELECT * FROM `group_names` WHERE `group_id`=" . $group_id . " AND `period_id`=" . $this->getCurrentPeriod();
                $results4 = DB::getInstance()->getfromBD($sql4);
        if(!$results4){
                    $sql41 = "INSERT INTO `group_names`(`group_id`, `period_id`, `group_name`, `course`) VALUES ('" . $group_id . "','" . $this->getCurrentPeriod() . "','" . $this->getCurrentGroupName($globalName)['name'] . "','" . $this->getCurrentGroupName($globalName)['course'] . "')";
            DB::getInstance()->executeQuery($sql41);
        }

        ////////////////////////////////////////
        /// Table01, Table05, Table06, Table08
            $sql00 = "SELECT * FROM `students` WHERE `group_id`='" . $group_id . "' AND `period_id`=" . $this->getCurrentPeriod() . " ORDER BY `student_fio`";
                $resultGroup = DB::getInstance()->getfromBD($sql00);
        for($t1 = 0; $t1 < count($resultGroup); $t1++){
            // Table01
                    $sql01 = "INSERT INTO `table01`(`tbl01_id`, `student_id`, `home_address`, `living_address`, `father_fio`, `father_tel`, `mother_fio`, `mather_tel`, `student_email`, `student_tel`, `group_id`, `period_id`) VALUES (NULL,'" . $resultGroup[$t1]['student_id'] . "',NULL, NULL, NULL, NULL, NULL,NULL, NULL, NULL, NULL, NULL)";
            DB::getInstance()->executeQuery($sql01);

            // Table05
                $t5arr = (object)["group"=>$group_id, "period"=>$this->getCurrentPeriod() ];
            $table05 = new Table05($t5arr);
            $table05->checkTable5ForActualData();

            // Table06
                $t6arr = (object)["group"=>$group_id, "period"=>$this->getCurrentPeriod() ];
            $table06 = new Table06($t6arr);
            $table06->checkTable6ForActualData();

            // Table08
            /*
                $t8arr = (object)["group"=>$group_id, "period"=>$this->getCurrentPeriod() ];
            $table08 = new Table08($t8arr);
            $table08->fillTableWithData();
            */

        } // end FOR()

    } // end FOR()

  } // end function getGroupsForCuratorFromASU2()

  protected function dataFigureOut($info_svodka)
  {
    $output = [];
     $teacher_asu_id = $info_svodka -> accounts[0]->id;
    $teacher_real_id = $info_svodka -> id;

    $output['success'] = 1;
    $output['teacher_asu_id'] = $teacher_asu_id;
    $output['teacher_real_id'] = $teacher_real_id;
```

```
        $output['teacher_fio'] = $info_svodka->model->lastName . " " . $info_svodka->model->firstName . " " . $info_svodka->model->secondName ;
        $output['teacher_email'] = $info_svodka->email;
                $output['teacher_title'] = $info_svodka->accounts[0]->title;

                                $teacher_id = $this->setDataAboutTeacher($output);
        $output['teacher_id'] = $teacher_id;
        $this->getGroupsForCuratorFromASU2($output);
        return $output;
    }

                                protected function checkIfGroupIsNotInStudyProcess($group_asu_id)
    {
                        $studentsArr = $this->mkr->getStudentsInGroup5($group_asu_id); //that's right!
                                            $students = json_decode($studentsArr['response']);
        if($students){
            return 0;
        } else {
            return 1;
        }

    }
}
```