

ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ І. І. МЕЧНИКОВА

(повне найменування закладу вищої освіти)

Факультет математики, фізики та інформаційних технологій

(повне найменування факультету)

Кафедра інформаційних технологій

(повна назва кафедри)

Кваліфікаційна робота

на здобуття ступеня вищої освіти «Бакалавр»

«Інтегрована платформа навчального моніторингу та комунікації для учасників освітнього процесу»

(тема кваліфікаційної роботи українською мовою)

«An integrated educational monitoring and communication platform for the educational process participants»

(тема кваліфікаційної роботи англійською мовою)

Виконав: здобувач заочної форми навчання спеціальності 122 Комп'ютерні науки

(код, назва спеціальності)

Освітня програма Комп'ютерні науки

(назва)

Новрузлу Рза Елшан огли

(прізвище, ім'я, по-батькові здобувача)

Керівник ст. викладач Вохменцева Т.Б.

(науковий ступінь, вчене звання, прізвище, ініціали)


(підпис)

Рецензент Клепатська В.В.

(науковий ступінь, вчене звання, прізвище, ініціали)

Рекомендовано до захисту:
Протокол засідання кафедри
Інформаційних технологій

№ 1 від 09 червня 2024 р.

Завідувачка кафедри


КАЗАКОВА Надія
(прізвище, ім'я)

Захищено на засіданні ЕК № 13,
протокол № 4 від 19 червня 2024 р.

Оцінка задовільно / Е / 65
(за національною шкалою/шкалою ECTS/ бали)

Голова ЕК


КОПИЧЕНКО Іван
(прізвище, ім'я)

Одеса 2024

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ	6
ВСТУП	7
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ І ПОСТАНОВКА ЗАВДАННЯ.....	9
1.1 Характеристика предметної області	9
1.1.1 Освітній процес	10
1.1.2 Навчальний моніторинг	11
1.1.3 Освітня комунікація	12
1.2 Аналіз існуючих систем навчального моніторингу та комунікації ...	13
1.2.1 Eddy	14
1.2.2 Єдина школа	16
1.2.3 Human Школа	18
1.2.4 Нові Знання.....	20
1.2.5 Моя школа	23
1.2.6 SMART школа	25
1.3 Формулювання мети та завдань дослідження	28
2 ВИБІР АРХІТЕКТУРИ СИСТЕМИ ТА ПРОГРАМНИХ ЗАСОБІВ РЕАЛІЗАЦІЇ.....	31
2.1 Вибір архітектури системи	31
2.1.1 Клієнт-серверна архітектура	31
2.1.2 Багаторівнева архітектура	32
2.1.3 Обґрунтування вибору архітектури	32
2.2 Вибір програмних засобів реалізації	33
2.2.1 Мови програмування	33
2.2.2 Системи керування базами даних	35
2.2.3 Фреймворки та бібліотеки	35
2.3 Обґрунтування вибору програмних засобів	36
3 ПРОЕКТУВАННЯ МЕРЕЖЕВОГО WEB-РЕСУРСУ	38
3.1. Моделювання даних	39

3.1.1. Атрибутивна модель	39
3.1.2. Об'єктно-орієнтована модель	42
3.2. Проектування інтерфейсу користувача	45
3.2.1. Вимоги до інтерфейсу користувача	45
3.2.2. Структура інтерфейсу користувача	46
3.2.3. Макетування інтерфейсу користувача	47
4 ПРАКТИЧНА РЕАЛІЗАЦІЯ МЕРЕЖЕВОГО WEB-РЕСУРСУ	48
4.1. Керівництво додатком користувач-студент системи	50
4.2. Керівництво додатком користувач-викладач системи	52
ВИСНОВКИ.....	53
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	54

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

- СНУ – система управління навчанням.
- N-tier архітектура – багаторівнева архітектура.
- СКБД – систем керування базами даних.
- SQL – Structured Query Language.
- NoSQL – not only SQL.
- ООМ – об'єктно-орієнтована модель.
- ООП – об'єктно-орієнтоване програмування.
- UX/UI – User Experience Design/User Interface.

ВСТУП

Сучасний освітній процес динамічно розвивається, потребуючи нових інструментів та технологій для забезпечення його ефективності. Інформаційні технології стають все більш невід'ємною частиною освіти, відкриваючи нові можливості для навчання, моніторингу та комунікації. Одним із перспективних напрямків розвитку інформаційних технологій в освіті є створення інтегрованих платформ навчального моніторингу та комунікації. Ці платформи мають на меті об'єднати в єдиному середовищі різні функції, необхідні для ефективного освітнього процесу, такі як:

- навчальний моніторинг;
- освітня комунікація;
- управління навчальним процесом;
- інструменти для спільного навчання.

Інтегровані платформи навчального моніторингу та комунікації мають ряд переваг перед традиційними методами роботи: підвищення ефективності навчання, покращення співпраці, зниження витрат, підвищення доступності.

Завданнями дипломної роботи є:

- проаналізувати існуючі системи навчального моніторингу та комунікації;
- розробити модель даних для інтегрованої платформи навчального моніторингу та комунікації;
- спроектувати інтерфейс користувача інтегрованої платформи навчального моніторингу та комунікації;
- розробити програмне забезпечення для інтегрованої платформи навчального моніторингу та комунікації.

Практична реалізація результатів дипломної роботи може бути здійснена в освітніх закладах різного рівня. Інтегрована платформа навчального моніторингу та комунікації може бути використана для: підвищення

ефективності навчання студентів, покращення комунікації між викладачами та студентами, оптимізації управління навчальним процесом, впровадження нових методів навчання.

Метою дипломної роботи є розробка інтегрованої платформи навчального моніторингу та комунікації, яка буде відповідати сучасним вимогам освітнього процесу.

Дипломна робота містить в собі 54 сторінки, 1 таблицю, 18 рисунків та 13 посилань.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ І ПОСТАНОВКА ЗАВДАННЯ

1.1 Характеристика предметної області

У сучасному освітньому середовищі велика увага приділяється впровадженню технологій для покращення якості навчання та ефективності комунікації між учасниками освітнього процесу. Інтегрована платформа навчального моніторингу та комунікації стає невід'ємною складовою сучасної освіти, яка сприяє покращенню якості освіти та забезпечує ефективну взаємодію між всіма учасниками освітнього процесу.

Ця предметна область охоплює широкий спектр функцій і можливостей, включаючи: моніторинг навчальних досягнень, забезпечення зв'язку між учасниками освітнього процесу, персоналізоване навчання, моніторинг прогресу і успішності, розвиток професійних навичок вчителів.

Моніторинг навчальних досягнень. Платформа дозволяє збирати та аналізувати дані про навчальні досягнення учнів на різних етапах навчання. Це допомагає вчителям та адміністрації школи адаптувати навчальні програми та методики для оптимізації процесу навчання [1].

Забезпечення зв'язку між учасниками освітнього процесу. Платформа створює зручний та ефективний канал комунікації між учнями, вчителями, батьками та адміністрацією навчального закладу. Це сприяє взаєморозумінню, обміну інформацією та співпраці всередині освітньої спільноти.

Персоналізоване навчання. Платформа надає можливість індивідуалізувати навчальний процес, враховуючи потреби та можливості кожного учня. Це дозволяє кожному учневі розвиватися відповідно до його унікальних можливостей.

Моніторинг прогресу і успішності. Платформа надає можливість стежити за прогресом учнів, відстежувати їхні досягнення та вчасно реагувати на будь-які труднощі чи проблеми, які можуть виникнути під час навчання.

Розвиток професійних навичок вчителів. Платформа може включати в себе інструменти для підвищення кваліфікації вчителів, обміну досвідом та співпраці між педагогами для постійного покращення методик навчання.

Узагальнюючи, інтегрована платформа навчального моніторингу та комунікації стає необхідним інструментом для сучасних освітніх закладів, сприяючи підвищенню якості навчання та створюючи сприятливі умови для розвитку кожного учня.

1.1.1 Освітній процес

Інтегрована платформа навчального моніторингу та комунікації для учасників освітнього процесу є важливим інструментом у сучасній освіті. Вона створює цифрове середовище, що об'єднує учнів, вчителів, батьків та адміністрацію навчального закладу з метою поліпшення якості навчання та забезпечення ефективної комунікації. У сучасному освітньому середовищі, де розвиток технологій набуває все більшого значення, інтегрована платформа стає необхідним інструментом для підтримки навчального процесу. Вона сприяє зручній взаємодії між учасниками навчання, дозволяючи їм обмінюватися інформацією, спілкуватися та співпрацювати [1].

Однією з ключових функцій цієї платформи є моніторинг навчальних досягнень. Вона дозволяє збирати та аналізувати дані про навчальні успіхи учнів, що допомагає вчителям адаптувати свої методи навчання та програми під конкретні потреби учнів. Крім того, інтегрована платформа сприяє підвищенню залученості батьків до освітнього процесу, надаючи їм можливість отримувати інформацію про навчальний прогрес своїх дітей та зв'язуватися з вчителями.

Платформа також створює умови для персоналізованого навчання, де кожен учень може розвиватися відповідно до своїх індивідуальних потреб та здібностей. Це дозволяє ефективніше використовувати час та ресурси

навчального закладу. Крім того, платформа надає адміністрації навчального закладу інструменти для аналізу даних та прийняття стратегічних рішень з покращення освітнього процесу.

Загалом, інтегрована платформа навчального моніторингу та комунікації стає необхідним інструментом для підтримки сучасного освітнього процесу, сприяючи покращенню якості навчання та забезпечуючи успішний розвиток кожного учасника освітнього середовища.

1.1.2 Навчальний моніторинг

Навчальний моніторинг – це процес систематичного збору, аналізу та інтерпретації даних щодо навчальних результатів учнів з метою оцінки їхнього прогресу та вдосконалення навчального процесу.

Інтегрована платформа навчального моніторингу надає учасникам освітнього процесу зручний і централізований інструмент для збору, обробки та аналізу даних про навчальний прогрес учнів. Вона дозволяє вчителям ефективно відстежувати навчальні досягнення кожного учня, ідентифікувати слабкі та сильні сторони в навчанні та вчасно реагувати на потреби кожного учня.

Основні функції платформи навчального моніторингу включають збір та аналіз даних про академічний прогрес учнів, візуалізацію даних у зручній формі для подальшого аналізу, надання звітності для вчителів, батьків та адміністрації школи, а також можливість встановлення індивідуальних навчальних цілей та моніторинг їх досягнення [2].

Завданням інтегрованої платформи навчального моніторингу є створення умов для систематичного та ефективного контролю навчальних досягнень учнів, забезпечення персоналізованого навчання та підвищення якості освіти. Вона має сприяти покращенню якості навчання, підвищенню мотивації учнів до навчання та підтримці співпраці між учасниками освітнього процесу.

1.1.3 Освітня комунікація

Інтегрована платформа навчального моніторингу та комунікації відіграє важливу роль у сприянні ефективній взаємодії між учасниками освітнього процесу. Вона створює цифровий простір, де вчителі, учні, батьки та адміністрація школи можуть з легкістю обмінюватися інформацією, спілкуватися та співпрацювати для досягнення спільних цілей.

Одним із важливих аспектів цієї платформи є сприяння зв'язку між учнями та вчителями. Вони можуть обговорювати навчальні теми, вирішувати питання, ділитися ідеями та взаємно навчатися. Це створює сприятливі умови для активної участі у навчальному процесі та підвищує мотивацію учнів до навчання [2].

Крім того, платформа надає можливість для спілкування між вчителями, що сприяє обміну досвідом, навчальними матеріалами та кращими практиками. Це дозволяє педагогам вдосконалювати свої методи навчання та підвищувати якість освіти в цілому.

Також важливою функцією платформи є залучення батьків до освітнього процесу. Вони можуть слідкувати за навчальним прогресом своїх дітей, отримувати повідомлення про події в школі та спілкуватися з вчителями для вирішення будь-яких питань, що стосуються навчання. Платформа також створює зручний канал зв'язку між вчителями та адміністрацією школи. Це дозволяє вчасно вирішувати організаційні питання, забезпечувати координацію робіт та підтримувати ефективне функціонування навчального закладу.

Узагальнюючи, інтегрована платформа навчального моніторингу та комунікації створює умови для плідної співпраці між учасниками освітнього процесу, що сприяє покращенню якості навчання та розвитку кожного учня.

1.2 Аналіз існуючих систем навчального моніторингу та комунікації

Аналіз існуючих систем навчального моніторингу та комунікації в освітній галузі виявляє ряд важливих аспектів, які можна врахувати при розробці інтегрованої платформи для учасників освітнього процесу.

По-перше, на сьогоднішній день існують різні системи навчального моніторингу, які в основному зорієнтовані на збір та аналіз даних про навчальні досягнення учнів. Багато з цих систем надають звітність для вчителів та адміністрації, допомагаючи їм в оцінці прогресу учнів та прийнятті рішень щодо подальшого навчання.

Проте, існуючі системи часто мають обмежені можливості щодо комунікації між учасниками освітнього процесу. Наприклад, вони можуть не мати засобів для ефективного спілкування між учнями та вчителями або обміну інформацією між батьками та адміністрацією школи.

Також важливо враховувати різноманітні потреби та особливості кожного учасника освітнього процесу. Наприклад, учні можуть мати різні стилі навчання та індивідуальні потреби, які потребують персоналізованого підходу. Батьки можуть бажати більшої взаємодії з вчителями та більшої участі у навчальному процесі своїх дітей [3].

Отже, в розробці інтегрованої платформи необхідно врахувати ці різноманітні потреби та забезпечити зручний та ефективний інструмент для навчального моніторингу та комунікації для всіх учасників освітнього процесу. Така платформа має сприяти покращенню якості навчання, підвищенню мотивації учнів та підтримці співпраці між усіма сторонами освітнього середовища.

В Україні існує декілька популярних систем ведення навчального моніторингу, які допомагають вчителям, учням та батькам відстежувати успішність навчання.

1.2.1 Eddy

«Eddy» – це хмарна платформа управління освітнім процесом, яка використовується в закладах загальної середньої освіти України. Система пропонує широкий спектр функцій, які допомагають вчителям, учням та батькам організувати та оптимізувати навчальний процес. Основні функції «Eddy»:

- ведення електронного щоденника. Вчителі можуть виставляти оцінки, домашні завдання, коментарі та зауваження в електронному щоденнику, який доступний учням та батькам;
- класний журнал. Вчителі можуть вести електронний класний журнал, де фіксуються теми уроків, присутність учнів, домашні завдання та інші дані;
- онлайн-уроки. Вчителі можуть проводити онлайн-уроки за допомогою інструментів для відеозв'язку, спільного доступу до екрану та чату;
- спілкування. Вчителі, учні та батьки можуть спілкуватися один з одним за допомогою чату, форумів та оголошень;
- аналітика. Вчителі можуть отримувати звіти про успішність учнів, відвідуваність та інші дані;
- бібліотека ресурсів. Вчителі та учні можуть отримувати доступ до бібліотеки ресурсів, яка включає навчальні матеріали, методичні рекомендації, електронні книги та інші матеріали.

Перевагами «Eddy» можна вважати:

- покращення організації навчального процесу. «Eddy» допомагає вчителям організувати свої плани уроків, завдання та оцінювання, що робить навчальний процес більш ефективним;
- підвищення доступності інформації. Учні та батьки мають доступ до інформації про успішність, завдання та інші аспекти навчального процесу в режимі реального часу;

- покращення комунікації. «Eddy» покращує комунікацію між вчителями, учнями та батьками;
- підвищення аналітики. «Eddy» надає вчителям доступ до даних про успішність учнів, відвідуваність та інші аспекти навчального процесу, що дозволяє їм приймати більш обґрунтовані рішення;
- доступність. «Eddy» доступний з будь-якого пристрою, який має доступ до Інтернету.

Недоліками «Eddy» можна вважати:

- вартість. «Eddy» є платною платформою, що може бути недоступним для деяких закладів освіти;
- складність. «Eddy» має широкий спектр функцій, що може бути складним для деяких користувачів;
- залежність від Інтернету. «Eddy» залежить від наявності Інтернету, що може бути проблемою в деяких регіонах.

Загалом, «Eddy» є потужною платформою управління освітнім процесом, яка може допомогти вчителям, учням та батькам організувати та оптимізувати навчальний процес [3].

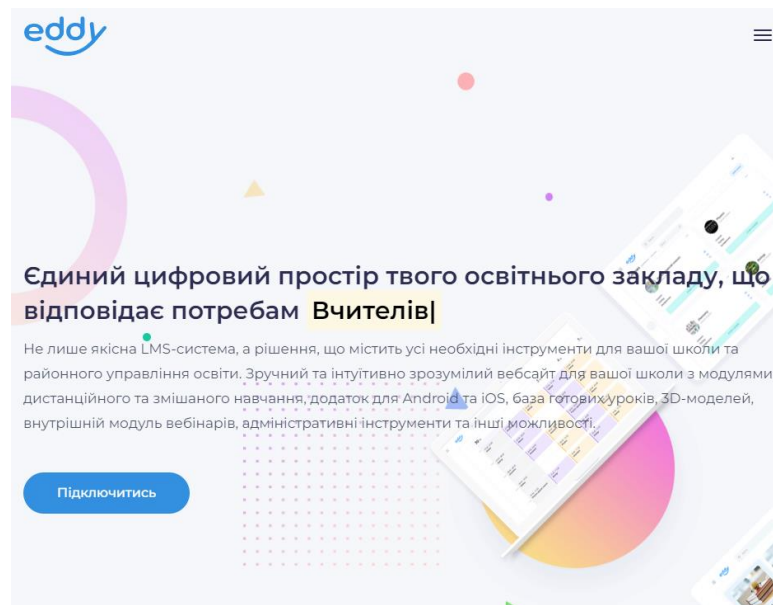


Рисунок 1.1 – Система «Eddy»

1.2.2 Єдина школа

«Єдина Школа» – це комплекс програмних рішень для управління освітнім процесом в Україні. Система використовується в закладах загальної середньої освіти, а також у деяких дошкільних та позашкільних закладах. Основні функції «Єдиної Школи» [4]:

- ведення електронного щоденника. Вчителі можуть виставляти оцінки, домашні завдання, коментарі та зауваження в електронному щоденнику, який доступний учням та батькам;
- класний журнал. Вчителі можуть вести електронний класний журнал, де фіксуються теми уроків, присутність учнів, домашні завдання та інші дані;
- розклад занять. Система автоматично генерує розклад занять для всього навчального закладу, враховуючи навантаження вчителів та кабінетів;
- відвідування. Батьки можуть відстежувати відвідування своїх дітей у школі в режимі реального часу;
- харчування. Система дозволяє організувати облік харчування учнів, включаючи замовлення їжі, оплату та облік харчових алергенів;
- бібліотека. Система надає доступ до електронної бібліотеки навчальних матеріалів;
- спілкування. Вчителі, учні та батьки можуть спілкуватися один з одним за допомогою чату, форумів та оголошень;
- аналітика. Система надає звіти про успішність учнів, відвідуваність, харчування та інші аспекти навчального процесу.

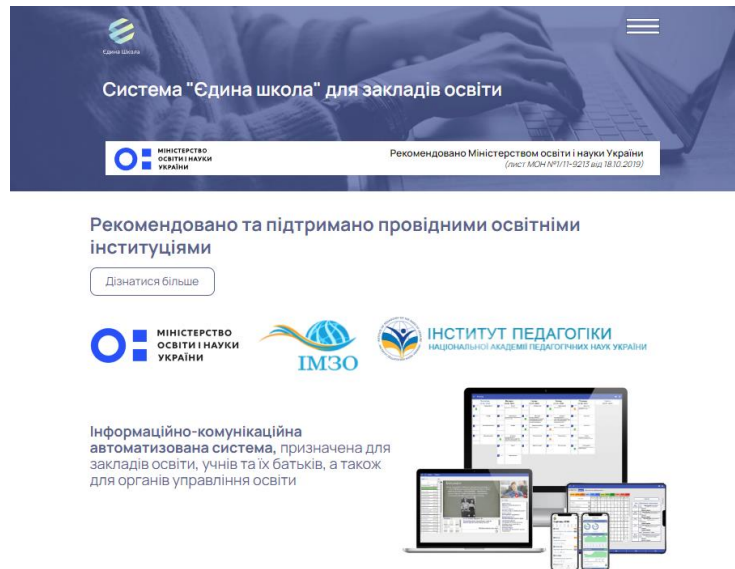


Рисунок 1.2 – Система «Єдина Школа»

Перевагами системи «Єдина Школа» є:

- покращення організації навчального процесу. Система допомагає вчителям організувати свої плани уроків, завдання та оцінювання, що робить навчальний процес більш ефективним;
- підвищення доступності інформації. Учні та батьки мають доступ до інформації про успішність, завдання та інші аспекти навчального процесу в режимі реального часу;
- покращення комунікації. Система покращує комунікацію між вчителями, учнями та батьками;
- підвищення аналітики. Система надає вчителям доступ до даних про успішність учнів, відвідуваність, харчування та інші аспекти навчального процесу, що дозволяє їм приймати більш обґрунтовані рішення;
- економія часу. Система автоматизує багато рутинних завдань, що дозволяє вчителям та адміністрації закладу економити час.

Недоліками системи «Єдина Школа» є:

- вартість. Система «Єдина Школа» є платною, що може бути недоступним для деяких закладів освіти [3];

- складність. Система має широкий спектр функцій, що може бути складним для деяких користувачів;
- залежність від Інтернету. Система залежить від наявності Інтернету, що може бути проблемою в деяких регіонах;
- можливі технічні проблеми. Як і будь-яка програмна система, «Єдина Школа» може мати технічні проблеми, які можуть призвести до перебоїв у роботі.

Загалом, «Єдина Школа» – це потужна платформа управління освітнім процесом, яка може допомогти вчителям, учням та батькам організувати та оптимізувати навчальний процес.

1.2.3 Human Школа

«HUMAN Школа» – це СНУ, розроблена компанією «HUMAN» для українських шкіл. Її мета – об'єднати всіх учасників освітнього процесу та допомогти вчителям та керівникам шкіл організувати внутрішні бізнес-процеси закладу освіти. Основні функції «HUMAN Школа» [4]:

- управління класом. Вчителі можуть створювати електронні щоденники, планувати уроки, виставляти оцінки, вести облік домашніх завдань, спілкуватися з батьками та учнями;
- управління школою. Адміністратори можуть вести електронний документообіг, генерувати звіти про успішність учнів, відвідуваність, харчування, контролювати роботу вчителів, організовувати батьківські збори та інші заходи;
- бібліотека ресурсів. Вчителі та учні мають доступ до бібліотеки, яка містить навчальні матеріали, методичні рекомендації, електронні книги та інші ресурси;
- спілкування. Вчителі, учні та батьки можуть спілкуватися один з одним за допомогою чату, форумів та оголошень;

- аналітика. Система надає дані про успішність учнів, відвідуваність, харчування та інші аспекти навчального процесу, що дозволяє приймати обґрунтовані рішення.

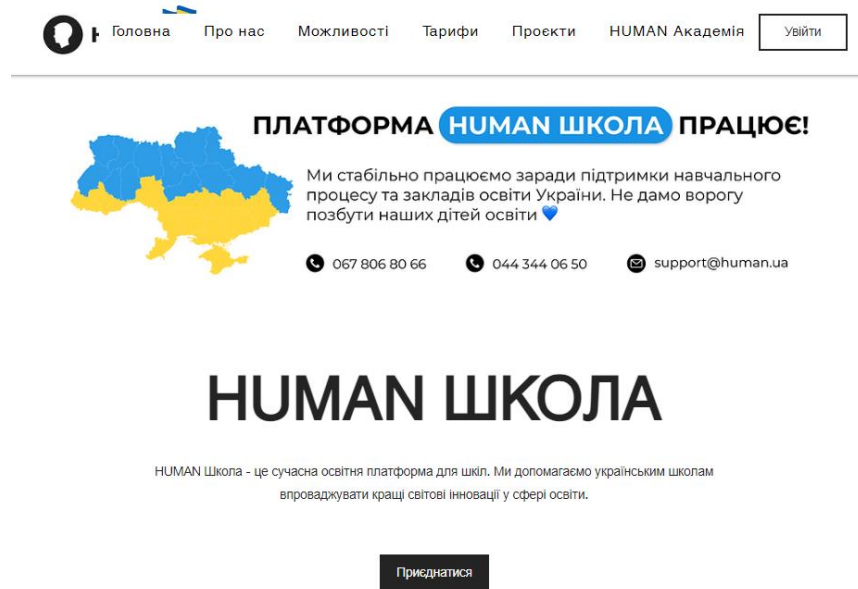


Рисунок 1.3 – Система «HUMAN Школа»

Перевагами системи «HUMAN Школа» є:

- покращення організації навчального процесу. Система допомагає вчителям та адміністрації організувати роботу школи, що робить навчальний процес більш ефективним [5];
- підвищення доступності інформації. Учні, батьки та вчителі мають доступ до інформації про успішність, завдання, розклад уроків, події школи та інші аспекти навчального процесу в режимі реального часу;
- покращення комунікації. Система покращує комунікацію між вчителями, учнями та батьками;
- підвищення аналітики. Система надає дані про успішність учнів, відвідуваність, харчування та інші аспекти навчального процесу, що дозволяє приймати обґрунтовані рішення;

- економія часу. Система автоматизує багато рутинних завдань, що дозволяє вчителям та адміністрації закладу економити час.

Недоліками системи «HUMAN Школа» є:

- вартість. Система «HUMAN Школа» є платною, що може бути недоступним для деяких закладів освіти;
- складність. Система має широкий спектр функцій, що може бути складним для деяких користувачів;
- залежність від Інтернету. Система залежить від наявності Інтернету, що може бути проблемою в деяких регіонах;
- можливі технічні проблеми. Як і будь-яка програмна система, «HUMAN Школа» може мати технічні проблеми, які можуть призвести до перебоїв у роботі [4].

Загалом, «HUMAN Школа» – це потужна платформа управління навчанням, яка може допомогти вчителям, учням та батькам організувати та оптимізувати навчальний процес.

1.2.4 Нові Знання

«Нові Знання» – це комплексна система управління освітнім процесом, розроблена компанією «Нові Знання» для українських шкіл. Її мета – допомогти вчителям, учням та батькам організувати та оптимізувати навчальний процес. Основні функції «Нових Знань»:

- електронний щоденник. Вчителі можуть виставляти оцінки, домашні завдання, коментарі та зауваження в електронному щоденнику, який доступний учням та батькам;
- класний журнал. Вчителі можуть вести електронний класний журнал, де фіксуються теми уроків, присутність учнів, домашні завдання та інші дані;

- розклад занять. Система автоматично генерує розклад занять для всього навчального закладу, враховуючи навантаження вчителів та кабінетів;
- спілкування. Вчителі, учні та батьки можуть спілкуватися один з одним за допомогою чату, форумів та оголошень;
- аналітика. Система надає звіти про успішність учнів, відвідуваність, харчування та інші аспекти навчального процесу;
- управління кадрами. Система дозволяє вести облік кадрів, вести таблиці робочого часу, організувати атестацію вчителів [5].

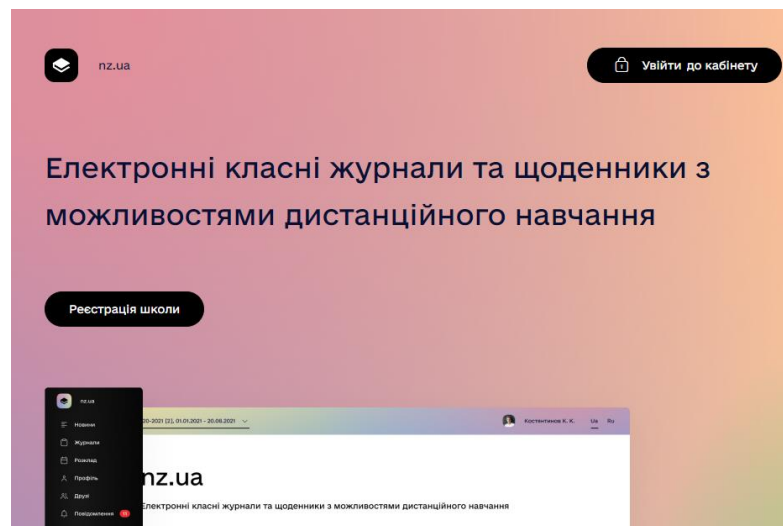


Рисунок 1.4 – Система «Нові Знання»

Переваги «Нових Знань»:

- покращення організації навчального процесу. Система допомагає вчителям організувати свої плани уроків, завдання та оцінювання, що робить навчальний процес більш ефективним;
- підвищення доступності інформації. Учні та батьки мають доступ до інформації про успішність, завдання, розклад уроків, події школи та інші аспекти навчального процесу в режимі реального часу;

- покращення комунікації. Система покращує комунікацію між вчителями, учнями та батьками;
- підвищення аналітики. Система надає дані про успішність учнів, відвідуваність, харчування та інші аспекти навчального процесу, що дозволяє приймати обґрунтовані рішення;
- економія часу. Система автоматизує багато рутинних завдань, що дозволяє вчителям та адміністрації закладу економити час;
- комплексність. Система «Нові Знання» охоплює всі аспекти управління освітнім процесом, що робить її зручним інструментом для адміністрації закладу освіти.
- вартість. Система «Нові Знання» є безоплатною.

Недоліки "Нових Знань":

- складність. Система має широкий спектр функцій, що може бути складним для деяких користувачів;
- залежність від Інтернету. Система залежить від наявності Інтернету, що може бути проблемою в деяких регіонах;
- можливі технічні проблеми. Як і будь-яка програмна система, «Нові Знання» може мати технічні проблеми, які можуть призвести до перебоїв у роботі [6].

Загалом, «Нові Знання» – це потужна платформа управління освітнім процесом, яка може допомогти вчителям, учням та батькам організувати та оптимізувати навчальний процес.

1.2.5 Моя школа

«Моя Школа» – це хмарна платформа управління освітнім процесом, розроблена для українських шкіл. Її мета – допомогти вчителям, учням та батькам організувати та оптимізувати навчальний процес. Основні функції «Моєї Школи» [5]:

- електронний щоденник: Вчителі можуть виставляти оцінки, домашні завдання, коментарі та зауваження в електронному щоденнику, який доступний учням та батькам.
- класний журнал. Вчителі можуть вести електронний класний журнал, де фіксуються теми уроків, присутність учнів, домашні завдання та інші дані;
- розклад занять. Система автоматично генерує розклад занять для всього навчального закладу, враховуючи навантаження вчителів та кабінетів;
- бібліотека ресурсів. Вчителі та учні мають доступ до бібліотеки, яка містить навчальні матеріали, методичні рекомендації, електронні книги та інші ресурси;
- спілкування. Вчителі, учні та батьки можуть спілкуватися один з одним за допомогою чату, форумів та оголошень;
- аналітика. Система надає звіти про успішність учнів, відвідуваність, харчування та інші аспекти навчального процесу;
- управління кадрами. Система дозволяє вести облік кадрів, вести таблиці робочого часу, організовувати атестацію вчителів;
- фінанси. Система дозволяє вести облік доходів та витрат, формувати звіти про фінансову діяльність закладу освіти.

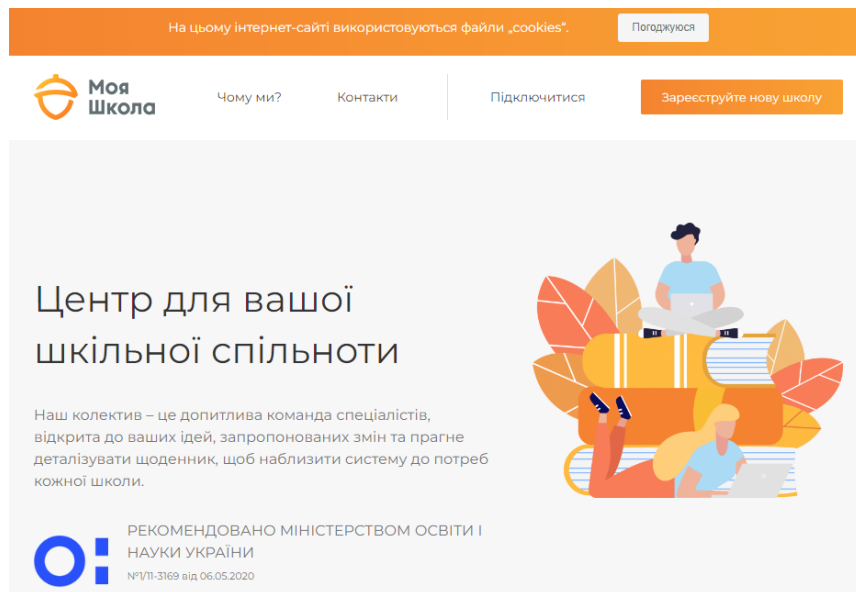


Рисунок 1.5 – Система «Моя Школа»

Переваги «Моєї Школи»:

- покращення організації навчального процесу. Система допомагає вчителям організувати свої плани уроків, завдання та оцінювання, що робить навчальний процес більш ефективним;
- підвищення доступності інформації. Учні та батьки мають доступ до інформації про успішність, завдання, розклад уроків, події школи та інші аспекти навчального процесу в режимі реального часу;
- покращення комунікації. Система покращує комунікацію між вчителями, учнями та батьками [6];
- підвищення аналітики. Система надає дані про успішність учнів, відвідуваність, харчування та інші аспекти навчального процесу, що дозволяє приймати обґрунтовані рішення;
- економія часу. Система автоматизує багато рутинних завдань, що дозволяє вчителям та адміністрації закладу економити час;
- комплексність. Система «Моя Школа» охоплює всі аспекти управління освітнім процесом, що робить її зручним інструментом для адміністрації закладу освіти.

Недоліки «Моєї Школи»:

- вартість. Система «Моя Школа» є платною, що може бути недоступним для деяких закладів освіти;
- складність. Система має широкий спектр функцій, що може бути складним для деяких користувачів;
- залежність від Інтернету. Система залежить від наявності Інтернету, що може бути проблемою в деяких регіонах;
- можливі технічні проблеми. Як і будь-яка програмна система, «Моя Школа» може мати технічні проблеми, які можуть призвести до перебоїв у роботі.

Загалом, «Моя Школа» – це потужна платформа управління освітнім процесом, яка може допомогти вчителям, учням та батькам організувати та оптимізувати навчальний процес.

1.2.6 SMART школа

«SMART школа» – це українська освітня інформаційна система, розроблена для загальноосвітніх шкіл. Її мета – допомогти вчителям, учням та батькам організувати та оптимізувати навчальний процес. Основні функції «SMART школи» [7]:

- електронний щоденник. Вчителі можуть виставляти оцінки, домашні завдання, коментарі та зауваження в електронному щоденнику, який доступний учням та батькам;
- класний журнал. Вчителі можуть вести електронний класний журнал, де фіксуються теми уроків, присутність учнів, домашні завдання та інші дані;
- розклад занять. Система автоматично генерує розклад занять для всього навчального закладу, враховуючи навантаження вчителів та кабінетів;

- бібліотека ресурсів. Вчителі та учні мають доступ до бібліотеки, яка містить навчальні матеріали, методичні рекомендації, електронні книги та інші ресурси;
- спілкування. Вчителі, учні та батьки можуть спілкуватися один з одним за допомогою чату, форумів та оголошень;
- аналітика. Система надає звіти про успішність учнів, відвідуваність, харчування та інші аспекти навчального процесу;
- управління кадрами. Система дозволяє вести облік кадрів, вести таблиці робочого часу, організувати атестацію вчителів;
- фінанси. Система дозволяє вести облік доходів та витрат, формувати звіти про фінансову діяльність закладу освіти [7];
- онлайн-уроки. Вчителі можуть проводити онлайн-уроки за допомогою інструментів для відеозв'язку, спільного доступу до екрану та чату.

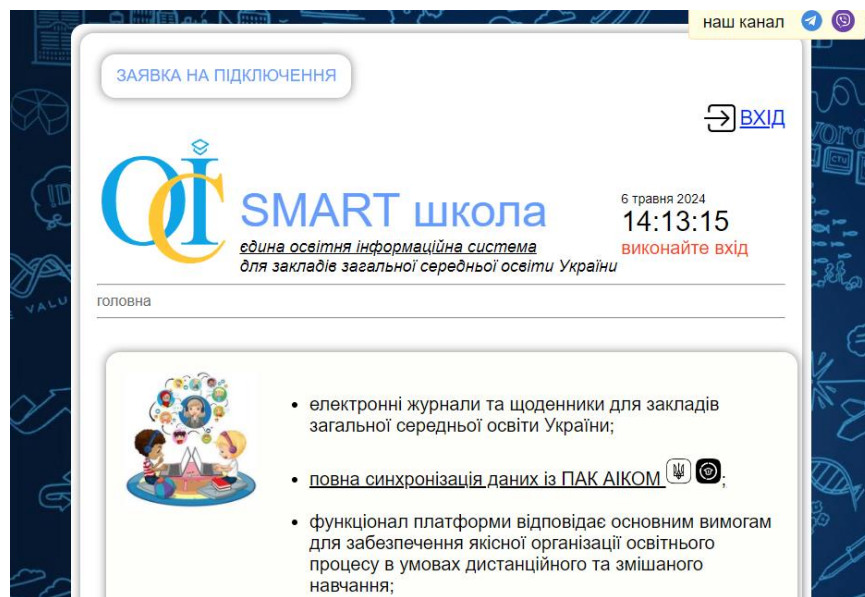


Рисунок 1.6 – Система «SMART школа»

Переваги «SMART школи»:

- покращення організації навчального процесу. Система допомагає вчителям організувати свої плани уроків, завдання та оцінювання, що робить навчальний процес більш ефективним;
- підвищення доступності інформації. Учні та батьки мають доступ до інформації про успішність, завдання, розклад уроків, події школи та інші аспекти навчального процесу в режимі реального часу;
- покращення комунікації. Система покращує комунікацію між вчителями, учнями та батьками;
- підвищення аналітики. Система надає дані про успішність учнів, відвідуваність, харчування та інші аспекти навчального процесу, що дозволяє приймати обґрунтовані рішення;
- економія часу. Система автоматизує багато рутинних завдань, що дозволяє вчителям та адміністрації закладу економити час;
- онлайн-навчання. Система дозволяє проводити онлайн-уроки, що може бути корисно для учнів, які не можуть бути присутніми на заняттях очно.

Недоліки «SMART школи» [8]:

- вартість. Система «SMART школа» є платною, що може бути недоступним для деяких закладів освіти;
- складність. Система має широкий спектр функцій, що може бути складним для деяких користувачів;
- залежність від Інтернету. Система залежить від наявності Інтернету, що може бути проблемою в деяких регіонах;
- можливі технічні проблеми. Як і будь-яка програмна система, «SMART школа» може мати технічні проблеми, які можуть призвести до перебоїв у роботі.

Загалом, «SMART школа» – це потужна платформа управління освітнім процесом, яка може допомогти вчителям, учням та батькам організувати та оптимізувати навчальний процес.

Після аналізу систем, перейдемо до порівняння, в таблиці 1 зображено резюме переваг та недоліків систем [8].

Таблиця 1 – Переваги та недоліки систем навчального моніторингу та комунікації.

	Покращення організації навчального процесу	Підвищення доступності інформації	Покращення комунікації	Підвищення аналітики	Доступність	Економія часу	Комплексність	Безкоштовність	Онлайн-навчання	Незалежність від Інтернету	Відсутність технічних проблем
Eddy	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✓
Єдина Школа	✓	✓	✓	✓	✗	✓	✗	✗	✗	✗	✗
Human Школа	✓	✓	✓	✓	✗	✓	✗	✗	✗	✗	✗
Нові Знання	✓	✓	✓	✓	✗	✓	✓	✓	✗	✗	✗
Моя Школа	✓	✓	✓	✓	✗	✓	✓	✗	✗	✗	✗
SMART школа	✓	✓	✓	✓	✗	✓	✓	✗	✓	✗	✗

У результаті порівняння, можемо сформулювати мету та завдання дослідження.

1.3 Формулювання мети та завдань дослідження

Мета розробки: створити єдину, зручну та функціональну платформу для комплексного моніторингу навчального процесу та ефективної комунікації між усіма учасниками освітнього процесу: вчителями, учнями, батьками та адміністрацією закладу освіти.

Завдання поділяється на декілька пунктів: моніторинг навчального процесу, комунікація, аналітика, економія часу, додаткові функції.

Моніторинг навчального процесу:

- розробка системи електронного щоденника, де вчителі зможуть виставляти оцінки, домашні завдання, коментарі та зауваження, доступні для учнів та батьків у режимі реального часу;
- створення електронного класного журналу, де буде фіксуватися інформація про теми уроків, присутність учнів, домашні завдання та інші дані;
- автоматизація генерації розкладу занять для всього навчального закладу, враховуючи навантаження вчителів та кабінетів;
- надання можливості відстежувати відвідування учнями занять у режимі реального часу;
- організація обліку харчування учнів, включаючи замовлення їжі, оплати та облік харчових алергенів;
- створення бібліотеки навчальних матеріалів, методичних рекомендацій, електронних книг та інших ресурсів, доступних для вчителів та учнів.

Комунікація:

- забезпечення зручних каналів спілкування між вчителями, учнями та батьками за допомогою чату, форумів та оголошень;
- надання можливості організації батьківських зборів та інших заходів онлайн.

Аналітика:

- формування звітів про успішність учнів, відвідуваність, харчування та інші аспекти навчального процесу, доступних для вчителів, адміністрації та батьків;
- надання можливості аналізувати дані для прийняття обґрунтованих рішень щодо покращення навчального процесу.

Економія часу:

- автоматизація рутинних завдань для вчителів та адміністрації закладу освіти;
- зменшення витрат часу на пошук та обмін інформацією.

Додаткові функції:

- розробка системи онлайн-уроків, що дозволить учням відвідувати заняття дистанційно;
- впровадження системи управління кадрами для ведення обліку кадрів, ведення табелів робочого часу, організації атестації вчителів;
- розробка системи фінансів для ведення обліку доходів та витрат, формування звітів про фінансову діяльність закладу освіти.

Очікувані результати:

- покращення організації та ефективності навчального процесу;
- підвищення доступності інформації для всіх учасників освітнього процесу;
- покращення комунікації та взаємодії між вчителями, учнями та батьками;
- економія часу та ресурсів для закладу освіти;
- підвищення рівня задоволеності учасників освітнього процесу.

Інтегрована платформа навчального моніторингу та комунікації має стати незамінним інструментом для всіх учасників освітнього процесу, сприяючи покращенню якості освіти та створенню сприятливого середовища для навчання та розвитку [9].

2 ВИБІР АРХІТЕКТУРИ СИСТЕМИ ТА ПРОГРАМНИХ ЗАСОБІВ РЕАЛІЗАЦІЇ

2.1 Вибір архітектури системи

При розробці інтегрованої платформи навчального моніторингу та комунікації для учасників освітнього процесу, вибір архітектури системи є ключовим аспектом, що впливає на функціональність, масштабованість, надійність та зручність користування. Для досягнення цих цілей розглянемо декілька популярних архітектурних підходів та оберемо найбільш оптимальний [10].

2.1.1 Клієнт-серверна архітектура

Клієнт-серверна архітектура є одним з найбільш поширених підходів у розробці інформаційних систем. Вона базується на чіткому розподілі функцій між клієнтами, які відповідають за інтерфейс взаємодії користувачів, та сервером, який обробляє запити, зберігає дані та виконує обчислення.

Основними компонентами є клієнт та сервер. Клієнт приймає вигляд веб-додатку або мобільного додатку, через який користувачі взаємодіють із системою. Сервер реалізується, як централізований вузол, який обробляє запити від клієнтів, виконує бізнес-логіку та керує базами даних.

Перевагами даної архітектури можна вважати: централізоване управління даними, що забезпечує їхню цілісність та безпеку; спрощення оновлення та підтримки системи, оскільки більшість змін вносяться на серверній стороні; можливість впровадження складних механізмів аутентифікації та авторизації.

Недоліками системи є: можливість перевантаження сервера при великій кількості одночасних запитів; висока залежність від мережевої інфраструктури, яка може стати "вузьким місцем" системи; обмежена масштабованість порівняно з більш сучасними архітектурними підходами.

2.1.2 Багаторівнева архітектура

N-tier архітектура представляє більш складний та гнучкий підхід, який розділяє систему на декілька рівнів (слоїв), кожен з яких відповідає за конкретний набір функцій. Найбільш поширена реалізація передбачає три рівні: презентаційний, логіки бізнесу та рівень даних.

Презентаційний рівень включає інтерфейс користувача, що може бути реалізований як веб-сторінки або мобільні додатки. Рівень логіки бізнесу це компоненти, що виконують бізнес-логіку, обробляють дані та приймають рішення. Рівень даних базується на базах даних та системах управління даними, що зберігають інформацію та забезпечують доступ до неї [11].

До переваг можна винести: чітке розмежування відповідальності між різними частинами системи, що спрощує її розробку та підтримку; покращену масштабованість завдяки можливості незалежного масштабування окремих рівнів; підвищену надійність та стійкість до збоїв, оскільки проблеми на одному рівні не обов'язково впливають на інші.

Недоліками даної архітектури можна вважати: більш складний процес розробки та тестування через необхідність координації між різними рівнями; збільшені вимоги до мережевої інфраструктури, оскільки кожен рівень взаємодіє з іншими через мережу; можливі додаткові затримки в обробці запитів через багаторівневу передачу даних.

2.1.3 Обґрунтування вибору архітектури

Для інтегрованої платформи навчального моніторингу та комунікації обидва розглянуті підходи мають свої переваги, проте багаторівнева архітектура є більш оптимальним вибором з причин: масштабованості, гнучкості, розширюваності та надійності.

Багаторівнева архітектура дозволяє масштабувати окремі компоненти системи незалежно один від одного, що особливо важливо для систем з

великою кількістю користувачів та високим навантаженням. Чітке розмежування між рівнями презентації, бізнес-логіки та даних дозволяє легко вносити зміни в одну частину системи без значного впливу на інші. Можливість додавання нових функцій та модулів без суттєвого перероблення існуючої архітектури забезпечує зручність подальшого розвитку системи. Відокремлення логіки бізнесу від рівня даних та презентаційного рівня підвищує стійкість системи до збоїв, оскільки проблеми в одному з компонентів менше впливають на інші [11].

Таким чином, багаторівнева архітектура найкраще відповідає потребам інтегрованої платформи навчального моніторингу та комунікації, забезпечуючи необхідну масштабованість, гнучкість, розширюваність та надійність.

2.2 Вибір програмних засобів реалізації

Вибір програмних засобів реалізації є критично важливим етапом у розробці інтегрованої платформи навчального моніторингу та комунікації. Правильний вибір мов програмування, СКБД, фреймворків та бібліотек безпосередньо впливає на ефективність, продуктивність, надійність та гнучкість майбутньої системи. У цьому розділі ми детально розглянемо ключові технологічні рішення, які будуть використані для створення платформи, а також обґрунтуємо їх вибір з огляду на специфічні вимоги проекту.

2.2.1 Мови програмування

Для розробки інтегрованої платформи навчального моніторингу та комунікації слід обрати мови програмування, які забезпечують високу продуктивність, зручність розробки та підтримку сучасних веб-технологій.

JavaScript є основною мовою для фронтенд-розробки, яка використовується для створення інтерфейсів користувача. Її популярність обумовлена широкою підтримкою всіма сучасними веб-браузерами, а також великою

кількістю бібліотек та фреймворків, які значно спрощують розробку динамічних веб-додатків.

TypeScript, як надмножина JavaScript, додає статичну типізацію, що покращує якість коду та знижує кількість помилок. Завдяки TypeScript розробники можуть виявляти помилки ще на етапі написання коду, що значно підвищує надійність та зручність підтримки великих проектів. Крім того, використання TypeScript сприяє кращій структурованості коду та спрощує інтеграцію з іншими інструментами і фреймворками [12].

Python використовується для бекенд-розробки завдяки своїй простоті, великій кількості бібліотек та фреймворків, а також підтримці об'єктно-орієнтованого та функціонального програмування. Він підходить для швидкого прототипування та реалізації складних логічних процесів. Python має чистий і зрозумілий синтаксис, що робить його одним з найулюбленіших мов розробників для створення веб-додатків.

Java забезпечує високу продуктивність та надійність, що робить її придатною для розробки масштабованих та багаторівневих систем. Вона дозволяє створювати високопродуктивні, масштабовані та надійні додатки. Java підтримує велику кількість інструментів та фреймворків, які сприяють швидкій та ефективній розробці програмних продуктів.

SQL використовується для взаємодії з СКБД і забезпечує роботу з реляційними базами даних. SQL дозволяє ефективно зберігати, витягувати та маніпулювати даними, забезпечуючи високу продуктивність та надійність операцій з базами даних.

2.2.2 Системи керування базами даних

Вибір СКБД залежить від вимог до зберігання, доступу та обробки даних.

PostgreSQL є реляційною СКБД, яка забезпечує високу надійність, відповідність стандартам SQL та розширюваність. Вона підходить для складних запитів та транзакцій, надаючи можливість ефективно працювати з великими обсягами даних.

MySQL є популярною реляційною СКБД з високою продуктивністю, яка забезпечує простоту у налаштуванні та використанні. Вона часто використовується у веб-додатках завдяки своїй стабільності та підтримці великої кількості користувачів.

MongoDB є NoSQL СКБД, яка підходить для зберігання великих обсягів даних, що не мають чіткої структури. Вона забезпечує високу масштабованість та гнучкість, що робить її ідеальною для сучасних веб-додатків з динамічними схемами даних [12].

2.2.3 Фреймворки та бібліотеки

Для прискорення розробки та забезпечення високої якості коду необхідно використовувати сучасні фреймворки та бібліотеки.

React є бібліотекою для побудови інтерфейсів користувача, яка забезпечує високу продуктивність та зручність у розробці завдяки компонентному підходу. Вона дозволяє створювати динамічні та інтерактивні користувацькі інтерфейси.

Angular є фреймворком для створення динамічних веб-додатків з повною підтримкою двостороннього зв'язку даних та модульності. Він надає потужні інструменти для побудови складних інтерфейсів користувача.

Vue.js є прогресивним фреймворком для побудови інтерфейсів користувача, який легко інтегрується у проекти будь-якого масштабу. Він поєднує в

собі простоту та гнучкість, дозволяючи швидко створювати ефективні веб-додатки.

Django є потужним фреймворком для Python, який забезпечує швидкий розвиток додатків завдяки вбудованим компонентам та адміністративному інтерфейсу. Він дозволяє створювати безпечні та масштабовані веб-додатки з мінімальними зусиллями.

Spring є фреймворком для Java, який підтримує створення масштабованих, надійних та безпечних додатків. Він забезпечує потужний набір інструментів для побудови складних бізнес-логік.

Express.js є мінімалістичним фреймворком для Node.js, який забезпечує створення серверної частини веб-додатків з високою продуктивністю. Він надає легкий та гнучкий підхід до розробки серверної логіки [12].

2.3 Обґрунтування вибору програмних засобів

Вибір програмних засобів є ключовим аспектом успішної реалізації інтегрованої платформи навчального моніторингу та комунікації. У цьому розділі буде детально розглянуто причини, з яких були обрані конкретні мови програмування, системи керування базами даних, фреймворки та бібліотеки. Обґрунтування вибору базується на аналізі технічних вимог проекту, потреб користувачів, а також сучасних тенденцій у галузі програмної інженерії. Такий підхід гарантує, що обрані технології забезпечать високу продуктивність, надійність та гнучкість платформи, сприяючи ефективному виконанню її функцій.

Мовами програмування були обрані:

JavaScript (TypeScript). Ця мова є стандартом для фронтенд-розробки, підтримується всіма сучасними браузерами і має велику кількість бібліотек та інструментів. Використання TypeScript зменшує кількість помилок завдяки

статичній типізації, що покращує надійність та зручність підтримки великих проектів.

Вибір Python для бекенд-розробки обґрунтований його простотою, популярністю та наявністю потужних фреймворків (наприклад, Django). Python підходить для швидкого прототипування та реалізації складних логічних процесів.

Використання Java дозволяє створювати високопродуктивні, масштабовані та надійні системи. Вона підходить для розробки складних багаторівневих додатків.

Системи керування базами даних були обрані:

Вибір PostgreSQL обумовлений її високою надійністю, масштабованістю та підтримкою складних SQL-запитів. Вона підходить для зберігання та обробки структурованих даних.

Використання MongoDB виправдане у випадках, коли необхідно зберігати великі обсяги неструктурованих даних або даних з динамічною схемою. Вона забезпечує високу гнучкість та масштабованість.

Фреймворками та бібліотеками були обрані:

Вибір React обґрунтований його популярністю, великою спільнотою розробників та ефективністю у створенні динамічних інтерфейсів користувача.

Використання Django для бекенд-розробки дозволяє швидко створювати повнофункціональні веб-додатки завдяки великій кількості вбудованих компонентів [13].

Вибір Express.js для серверної частини веб-додатків надає можливість створювати високопродуктивні додатки з мінімальними накладними витратами.

Таким чином, обрані програмні засоби забезпечують високу продуктивність, гнучкість, надійність та зручність розробки інтегрованої платформи навчального моніторингу та комунікації.

3 ПРОЕКТУВАННЯ МЕРЕЖЕВОГО WEB-РЕСУРСУ

Процес проектування мережевого веб-ресурсу є критично важливим етапом створення інтегрованої платформи навчального моніторингу та комунікації. Він включає в себе визначення структури даних, вибір відповідних моделей даних та їх атрибутів, а також розробку механізмів для ефективного управління цими даними. На цьому етапі закладаються основи для майбутньої продуктивності, масштабованості та надійності системи.

У цьому розділі ми розглянемо основні моделі даних, які будуть використовуватись для забезпечення ефективного зберігання, обробки та доступу до інформації на платформі. Правильна організація даних дозволяє оптимізувати запити, зменшити затримки доступу до даних і забезпечити високу продуктивність системи. Ми зосередимося на двох основних моделях даних, які є найбільш поширеними в сучасних веб-додатках: атрибутивну модель та об'єктно-орієнтовану модель.

Атрибутивна модель базується на реляційних базах даних і забезпечує структуроване зберігання даних у вигляді таблиць. Ця модель дозволяє легко здійснювати запити до даних, використовуючи мову SQL, що є стандартом для реляційних баз даних. Атрибутивна модель є ефективною для систем з великим обсягом структурованих даних та складними запитами [13].

Об'єктно-орієнтована модель, з іншого боку, відображає дані у вигляді об'єктів, що дозволяє більш природньо моделювати реальні сутності та їх взаємодії. Ця модель є особливо корисною для систем, які використовують об'єктно-орієнтовані мови програмування, оскільки вона забезпечує пряме відображення об'єктів в коді на об'єкти в базі даних. Об'єктно-орієнтована модель підтримує складні зв'язки між об'єктами та дозволяє легко розширювати систему.

Таким чином, вибір відповідної моделі даних є ключовим фактором у забезпеченні успішної роботи платформи. У наступних підрозділах ми детально розглянемо атрибутивну та об'єктно-орієнтовану моделі, їх особливості,

переваги та недоліки, а також приклади їх застосування у контексті інтегрованої платформи навчального моніторингу та комунікації.

3.1. Моделювання даних

Моделювання даних є важливим аспектом проектування веб-ресурсу, оскільки правильна організація даних дозволяє забезпечити ефективне управління інформацією, оптимізувати запити та зменшити затримки доступу до даних. Важливість цього процесу полягає в тому, що добре структуровані дані сприяють підвищенню продуктивності системи, полегшують підтримку та розширення функціоналу, а також забезпечують високу надійність та безпеку даних.

У даному підрозділі ми розглянемо дві основні моделі даних: атрибутивну модель та об'єктно-орієнтовану модель. Кожна з цих моделей має свої переваги та підходить для різних типів завдань і систем, тому їх детальне вивчення допоможе визначити найкращий підхід для конкретного проекту.

3.1.1. Атрибутивна модель

Атрибутивна модель даних є однією з найпоширеніших методик моделювання, що використовується для визначення структури та властивостей даних у реляційних базах даних. Ця модель фокусується на визначенні атрибутів (властивостей) кожного об'єкта (сутності), а також на зв'язках між цими об'єктами. Основними компонентами атрибутивної моделі є сутності, атрибути та зв'язки.

Сутності – визначають основні об'єкти, які будуть зберігатися у базі даних. У контексті платформи навчального моніторингу сутностями можуть бути студенти, курси, викладачі та оцінки.

Атрибути – описують властивості кожної сутності. Атрибутами сутності "студент" можуть бути ім'я, прізвище, дата народження, електронна пошта.

Для сутності "курс" атрибутами можуть бути назва курсу, код курсу, опис та кількість кредитів. Викладачі можуть мати атрибути, такі як ім'я, прізвище, кафедра та контактна інформація. Оцінки можуть мати атрибути, такі як значення оцінки, дата виставлення оцінки та курс, до якого відноситься оцінка.

Зв'язки – визначають відносини між сутностями. Зв'язок між студентами та курсами може бути визначений як "студент записаний на курс". Це може бути реалізовано за допомогою таблиці, яка містить зовнішні ключі до таблиць студентів та курсів, що відображає зв'язок між цими сутностями. Наприклад, студент може бути зареєстрований на декілька курсів, а курс може мати декілька студентів, зареєстрованих на нього [12].

Атрибутивна модель дозволяє чітко визначити структуру бази даних, забезпечуючи зрозумілість та логічність зберігання інформації. Використання цієї моделі спрощує процес створення запитів до бази даних та забезпечує високу продуктивність системи за рахунок оптимізації структури даних.

Основні компоненти атрибутивної моделі.

Сутності:

- студенти: ім'я, прізвище, дата народження, електронна пошта;
- курси: назва курсу, код курсу, опис, кількість кредитів;
- викладачі: ім'я, прізвище, кафедра, контактна інформація;
- оцінки: значення оцінки, дата виставлення оцінки, курс.

Атрибути:

- кожна сутність має набір атрибутів, які описують її властивості;
- приклади атрибутів для сутності "студент": ім'я, прізвище, дата народження, електронна пошта;
- атрибути для сутності "курс": назва курсу, код курсу, опис, кількість кредитів;

Зв'язки:

- визначення відносин між сутностями, наприклад, зв'язок між студентами та курсами;

- студент може бути зареєстрований на декілька курсів, а кожен курс може мати декілька студентів;
- для реалізації цього зв'язку використовується таблиця, яка містить зовнішні ключі до таблиць студентів та курсів.

Переваги атрибутивної моделі: чітка структура, простота запитів, оптимізація продуктивності, гнучкість.

Всі дані організовані у вигляді таблиць, що забезпечує логічне та структуроване зберігання інформації. Використання SQL для створення та виконання запитів до бази даних. Можливість оптимізації структури бази даних для підвищення продуктивності запитів. Легкість у додаванні нових атрибутів та сутностей у міру розвитку системи.

Атрибутивна модель (рис. 3.1) є потужним інструментом для організації даних у реляційних базах даних, що забезпечує ефективне управління інформацією та високий рівень продуктивності системи.

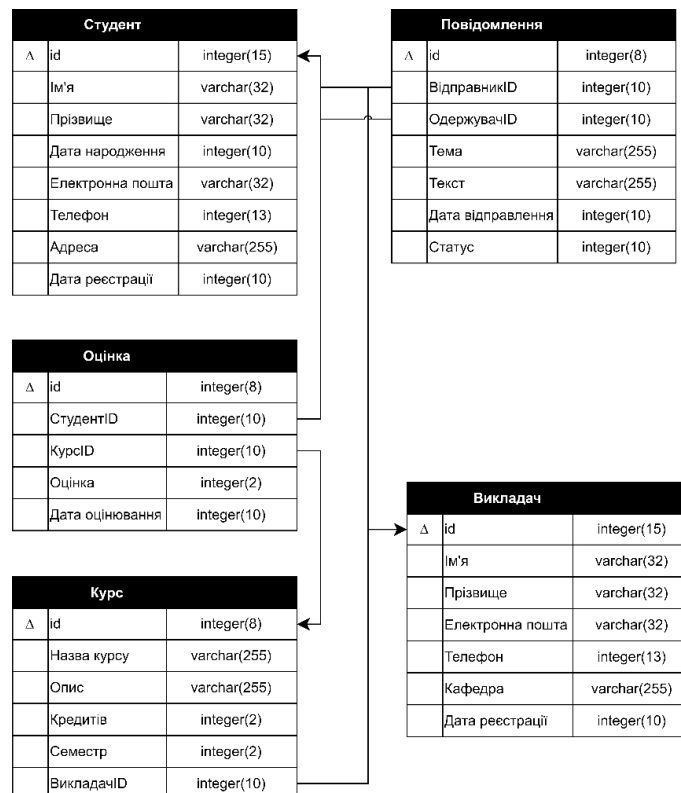


Рисунок 3.1 – Атрибутивна модель

3.1.2. Об'єктно-орієнтована модель

ООМ даних базується на принципах ООП і використовується для моделювання даних у вигляді об'єктів, які мають властивості та методи. Ця модель є більш гнучкою та масштабованою, особливо для складних систем з великою кількістю взаємозв'язаних об'єктів. Основні компоненти об'єктно-орієнтованої моделі: класи, об'єкти, наслідування, інкапсуляція, поліморфізм.

Визначають типи об'єктів, які будуть створені у системі. Клас містить атрибути (властивості) та методи (функції), які описують поведінку об'єктів цього типу. Є екземплярами класів і представляють конкретні сутності з реального світу. Кожен об'єкт має свій стан (значення атрибутів) та поведінку (методи). Дозволяє створювати нові класи на основі існуючих, успадковуючи їх властивості та методи. Це сприяє повторному використанню коду та зменшує його дублювання. Забезпечує захист даних, дозволяючи доступ до них лише через визначені методи. Дозволяє використовувати об'єкти різних класів через однаковий інтерфейс, що спрощує розробку та підтримку системи [11].

Об'єктно-орієнтована модель є ефективною для розробки складних і масштабованих систем, оскільки дозволяє організувати код у вигляді взаємозв'язаних об'єктів, що полегшує його розуміння та підтримку.

Клас "Користувач"

Базовий клас, який містить спільні атрибути для студентів та викладачів.

```
class User:
    def __init__(self, user_id, first_name, last_name, email, phone):
        self.user_id = user_id
        self.first_name = first_name
        self.last_name = last_name
        self.email = email
        self.phone = phone
```

Клас "Студент"

Наслідує клас "Користувач" і додає специфічні атрибути для студентів.

```
class Student(User):
```



```

def __init__(self, user_id, first_name, last_name, email, phone,
              birth_date, address, registration_date):
    super().__init__(user_id, first_name, last_name, email, phone)
    self.birth_date = birth_date
    self.address = address
    self.registration_date = registration_date
    self.courses = []

```

Клас "Викладач"

Наслідує клас "Користувач" і додає специфічні атрибути для викладачів.

```

class Teacher(User):
    def __init__(self, user_id, first_name, last_name, email, phone,
                  department, registration_date):
        super().__init__(user_id, first_name, last_name, email, phone)
        self.department = department
        self.registration_date = registration_date
        self.courses = []

```

Клас "Курс"

Містить атрибути курсу та посилання на викладача.

```

class Course:
    def __init__(self, course_id, name, description, credits, semester,
                  teacher):
        self.course_id = course_id
        self.name = name
        self.description = description
        self.credits = credits
        self.semester = semester
        self.teacher = teacher
        self.students = []

```

Клас "Оцінка"

Містить атрибути оцінки та посилання на студента і курс.

```

class Grade:
    def __init__(self, grade_id, student, course, value, date):
        self.grade_id = grade_id
        self.student = student
        self.course = course

```

```
self.value = value
self.date = date
```

Клас "Повідомлення"

Містить атрибути повідомлення та посилання на відправника і одержувача.

```
class Message:
    def __init__(self, message_id, sender, receiver, subject, text,
                 date_sent, status):
        self.message_id = message_id
        self.sender = sender
        self.receiver = receiver
        self.subject = subject
        self.text = text
        self.date_sent = date_sent
        self.status = status
```

Відносини між класами (рис. 3.2). Клас "Студент" має відношення "багато-до-багатьох" з класом "Курс" через атрибут courses. Клас "Викладач" має відношення "один-до-багатьох" з класом "Курс" через атрибут courses. Клас "Курс" має відношення "багато-до-одного" з класом "Викладач" через атрибут teacher. Клас "Оцінка" має відношення "один-до-одного" з класом "Студент" та класом "Курс". Клас "Повідомлення" має відношення "багато-до-багатьох" з класом "Користувач" через атрибути sender та receiver.

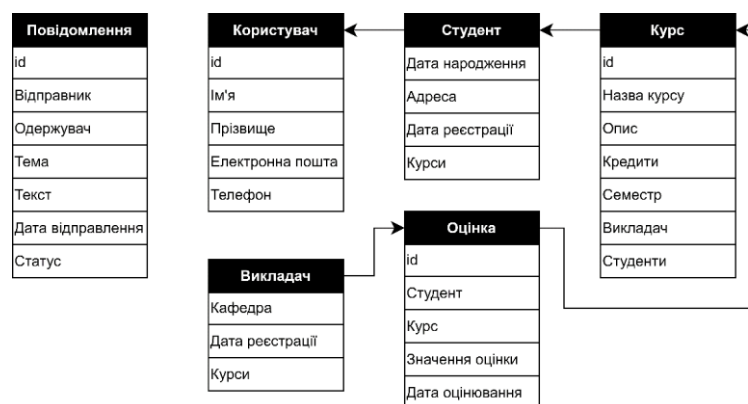


Рисунок 3.2 – Діаграма класів

Ця об'єктно-орієнтована модель відображає структуру та зв'язки між основними сутностями платформи навчального моніторингу та комунікації, забезпечуючи гнучкість, масштабованість та зручність підтримки системи.

3.2. Проектування інтерфейсу користувача

Під час проектування інтерфейсу користувача для інтегрованої платформи навчального моніторингу та комунікації важливо враховувати потреби користувачів та забезпечити зручність взаємодії з системою. У цьому розділі описані вимоги до інтерфейсу користувача, його структури та процесів макетування.

3.2.1. Вимоги до інтерфейсу користувача

Інтерфейс користувача повинен бути зручним, інтуїтивно зрозумілим та ефективним для користувачів з різними рівнями технічної підготовки. Для цього варто врахувати такі вимоги: зручність використання, інтуїтивність, підтримка різних пристроїв, естетика та дизайн [10].

Інтерфейс повинен бути легким у використанні та навігації, незалежно від того, чи є користувач новачком, чи досвідченим користувачем. Це досягається за рахунок зрозумілих і чітких інструкцій, простого доступу до основних функцій та можливості швидко переходити між розділами системи.

Елементи управління повинні бути розташовані логічно та інтуїтивно зрозуміло, щоб користувачам не було важко знаходити необхідні функції. Використання стандартних іконок та кнопок, відповідно до загальноприйнятих норм UX/UI дизайну, допоможе зменшити криву навчання для нових користувачів.

Інтерфейс повинен бути адаптований для роботи на різних пристроях, таких як комп'ютери, планшети та смартфони. Це означає, що всі елементи інтерфейсу повинні коректно відображатися та функціонувати на різних

розмірах екрану, забезпечуючи користувачам безперешкодний доступ до платформи з будь-якого пристрою.

Важливо, щоб інтерфейс був привабливим та естетичним, але при цьому не перевантаженим зайвими деталями. Використання простих і чистих дизайнів, гармонійних кольорових схем та відповідних шрифтів забезпечить приємний користувацький досвід і сприятиме тривалому залученню користувачів до платформи.

3.2.2. Структура інтерфейсу користувача

Структура інтерфейсу повинна бути логічною та зручною для користувача. Основні складові структури інтерфейсу можуть включати: головне меню, панель навігації, вмістова область, контекстні меню та інструменти.

Меню, яке надає доступ до основних функцій системи, таких як перегляд курсів, відправлення повідомлень, перегляд оцінок тощо. Головне меню повинно бути легко доступним з будь-якої сторінки платформи.

Розташована зазвичай по боці екрану і містить швидкі посилання на різні розділи або функції системи. Панель навігації може містити розділи для курсів, повідомлень, профілю користувача та інших важливих функцій.

Місце, де відображається основний вміст, такий як список курсів, деталі курсу, повідомлення тощо. Вмістова область повинна бути максимально просторою та не перевантаженою зайвими елементами, щоб користувачі могли зосередитися на основній інформації [9].

Меню або панель інструментів, які з'являються під час виконання певних дій або взаємодії з елементами інтерфейсу. Ці елементи повинні бути логічно впорядковані та легко доступні для швидкого виконання завдань.

3.2.3. Макетування інтерфейсу користувача

Макетування інтерфейсу полягає в створенні концептуальних макетів, які відображають структуру та вигляд інтерфейсу. Під час макетування важливо: розташування елементів, співвідношення, кольорова гамма та шрифти, відгуки користувачів.

Розмістіть елементи інтерфейсу так, щоб вони були зручно доступними та легко розпізнаваними. Наприклад, основні функції повинні бути на видному місці, а менш важливі – в окремих розділах або підменю.

Забезпечте правильні співвідношення між елементами, щоб уникнути перевантаження інтерфейсу та забезпечити гармонійний вигляд. Наприклад, текстові блоки повинні мати достатньо простору навколо себе для кращого сприйняття.

Використовуйте приємні для очей кольори та чіткі, читабельні шрифти. Колірна гамма повинна бути гармонійною, без надмірної кількості яскравих кольорів, які можуть відволікати увагу користувачів.

Залучайте користувачів до тестування макетів та збирайте їхні відгуки для подальшого вдосконалення інтерфейсу. Це допоможе виявити потенційні проблеми та зробити інтерфейс більш зручним та ефективним.

Ретельне проектування інтерфейсу користувача забезпечить зручність та ефективність використання інтегрованої платформи навчального моніторингу та комунікації. Врахування вимог до інтерфейсу, логічна структура та продумане макетування допоможуть створити інтуїтивно зрозумілий та естетично привабливий інтерфейс, який задовольнить потреби користувачів.

4 ПРАКТИЧНА РЕАЛІЗАЦІЯ МЕРЕЖЕВОГО WEB-РЕСУРСУ

Першою сторінкою нас зустрічає сторінка «Ласкаво просимо» (рис. 4.1), на даному екрані розміщуються текстові поля для вводу даних, кнопки «Увійти» та «Забули пароль». Після успішного введення даних, користувач потрапляє на сторінку курси (рис. 4.5).

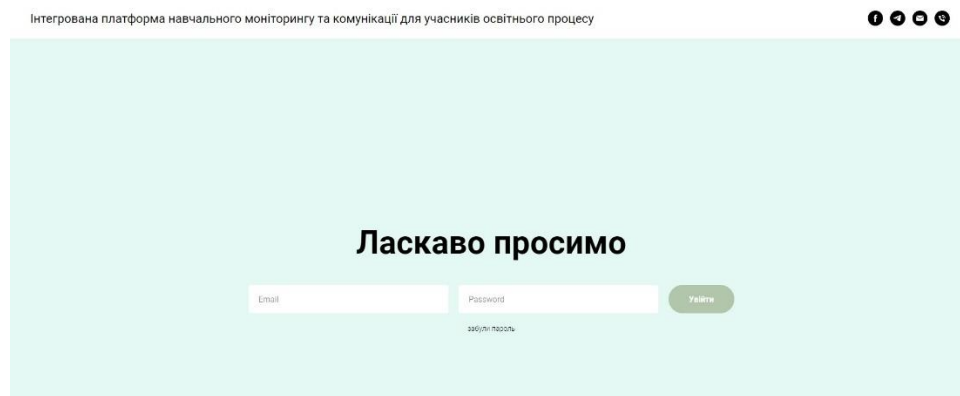


Рисунок 4.1 – Сторінка «Ласкаво просимо»

При не коректному введенні інформації, у користувача є можливість відновити пароль при натисканні на кнопку «Забули пароль», дана сторінка перезавантажиться та направить користувача на наступний екран (рис. 4.2).

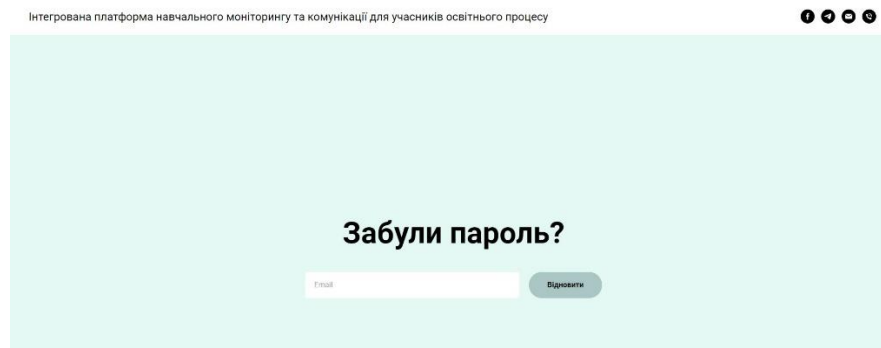


Рисунок 4.2 – Сторінка «Забули пароль»

Після приєднання студента або викладача до системи, перед ним з'явиться меню, яке зображене на рисунку 4.3. В даному меню є такі пункти, як «Курси» з пунктами другого рівня, тобто найменування саме які курси, меню «Повідомлення» та «Особистий кабінет».



Рисунок 4.3 – Сторінка «Курси», елемент «Меню»

При виборі пункту особистий кабінет користувач має змогу змінити контактну інформацію (рис. 4.4), таку як: електронна пошта, пароль, телефон та адресу за бажанням. Інформація що стосується Імені та дати народження не змінна самостійно, тільки через користувача адміністратора системи.

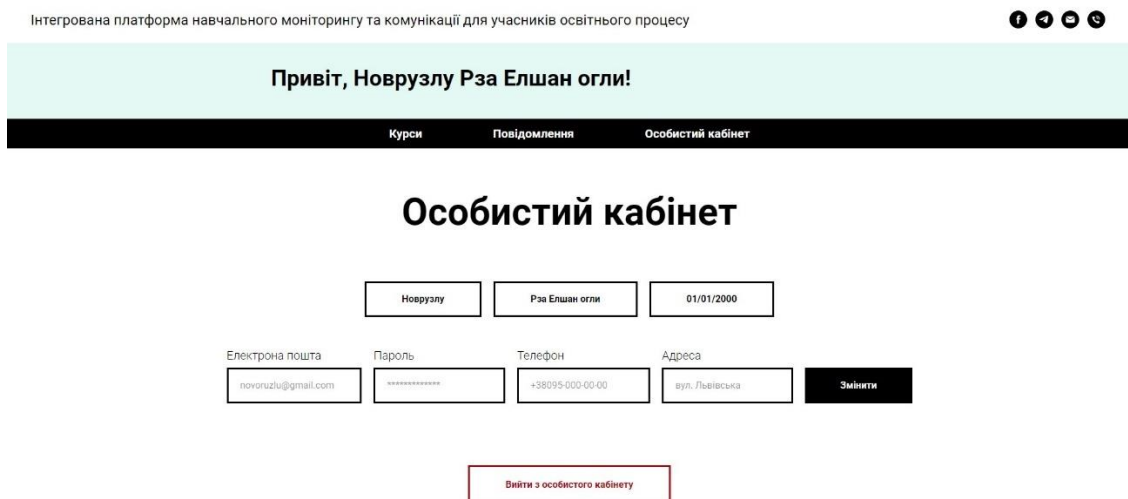
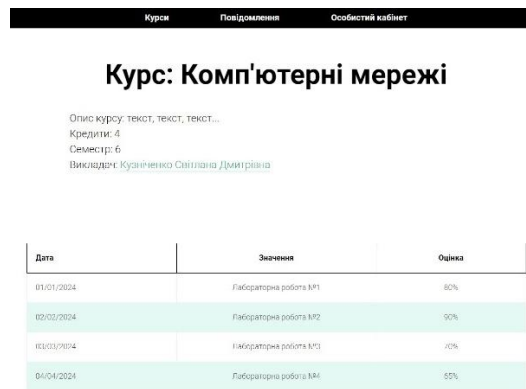


Рисунок 4.4 – Сторінка «Особистий кабінет»

4.1. Керівництво додатком користувач-студент системи

Користувач студент при виборі необхідного курсу потрапляє на відповідну сторінку (рис. 4.5). На даній сторінці зберігається інформація щодо дисципліни: опис курсу, кредити, семестр, викладач з посиланням на його сторінку (рис. 4.6). Також на сторінці знаходиться таблиця з інформацією про оцінювання даного студента викладачем, в якій вказано, за яку тему або освітню діяльність була виставлена необхідна оцінка.



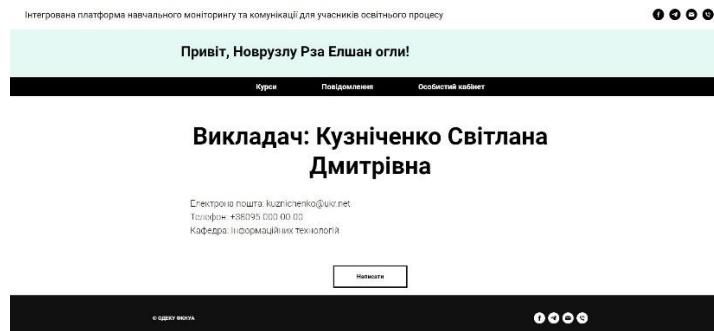
Курс: Комп'ютерні мережі

Опис курсу: текст, текст, текст...
 Кредити: 4
 Семестр: 6
 Викладач: Кузнiченко Світлана Дмитрiвна

Дата	Значення	Оцiнка
01/01/2024	Лабораторна робота М1	80%
02/02/2024	Лабораторна робота М2	90%
03/03/2024	Лабораторна робота М3	75%
04/04/2024	Лабораторна робота М4	55%

Рисунок 4.5 – Сторінка «Курс» для користувача студент

Сторінка викладача (рис. 4.6), на яку може перейти виконавець студент зберігає в собі інформацію щодо контактних даних викладача, таких як: електронна пошта, номер телефону та кафедра.



Інтегрована платформа навчального моніторингу та комунікації для учасників освітнього процесу

Привіт, Новрузлу Рза Елшан огли!

Курс: Комп'ютерні мережі

Викладач: Кузнiченко Світлана Дмитрiвна

Електронна пошта: kuznichenko@uk.net
 Телефон: +38095 000 00 00
 Кафедра: Інформаційні технології

Написати

© 2024. ВУЖТ

Рисунок 4.6 – Сторінка «Викладач» для користувача студент

Наступний екран – сторінка «Повідомлення», на даній сторінці студент має змогу подивитись усі активні чати з викладачами (рис. 4.7).

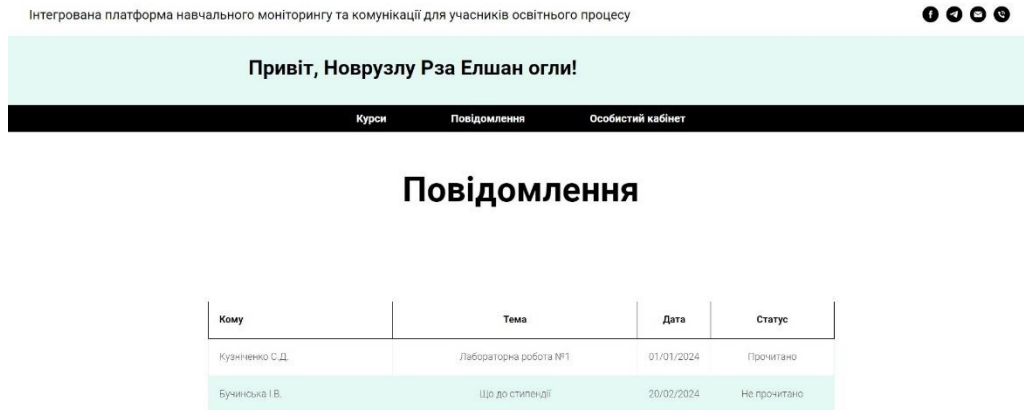


Рисунок 4.7 – Сторінка «Повідомлення» для користувача студент

Щоб створити новий чат для спілкування, потрібно натиснути кнопку написати на рисунку 4.6 та буде створено новий чат, а при натисканні на активний чат на рисунку 4.7 буде відкрито чат з цією особою, як показано на рисунку 4.8.

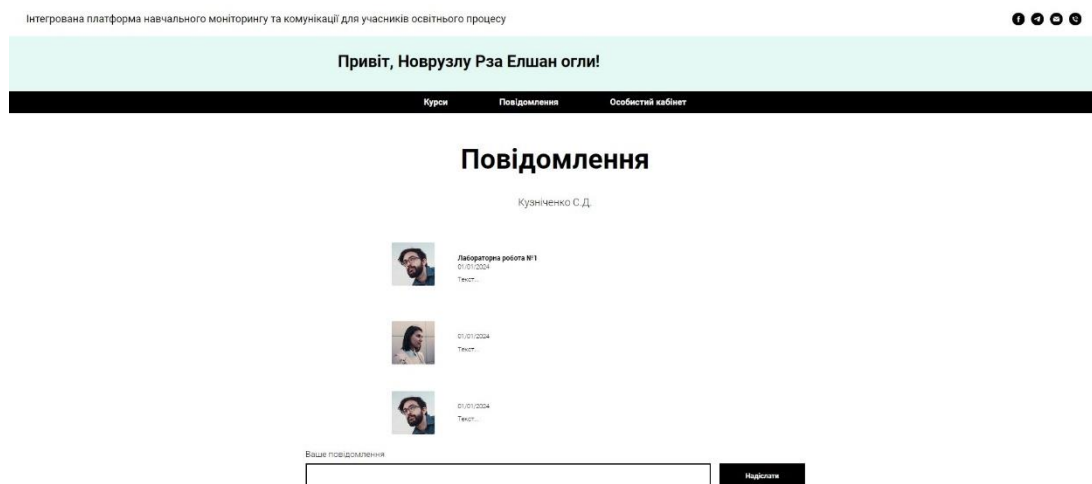


Рисунок 4.8 – Сторінка «Повідомлення», чат

4.2. Керівництво додатком користувач-викладач системи

Як і у користувача-студента, у викладача є можливість змінювати свої контактні дані на вкладці особистий кабінет. Вигляд «Курсу» має інший зовнішній вигляд (рис. 4.9), на даному екрані користувач-викладач має змогу не тільки виставляти оцінки студентам, а й змінювати інформацію щодо курсу.

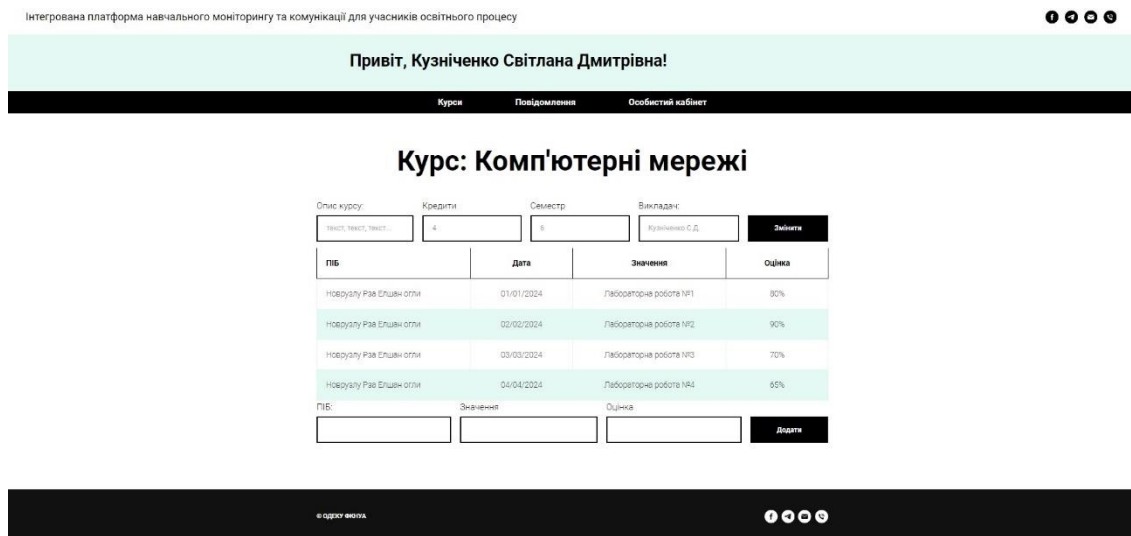


Рисунок 4.9 – Сторінка «Курс» для користувача викладач

Сторінка повідомлення виглядає так само, як і для користувача-студента (рис. 4.10).

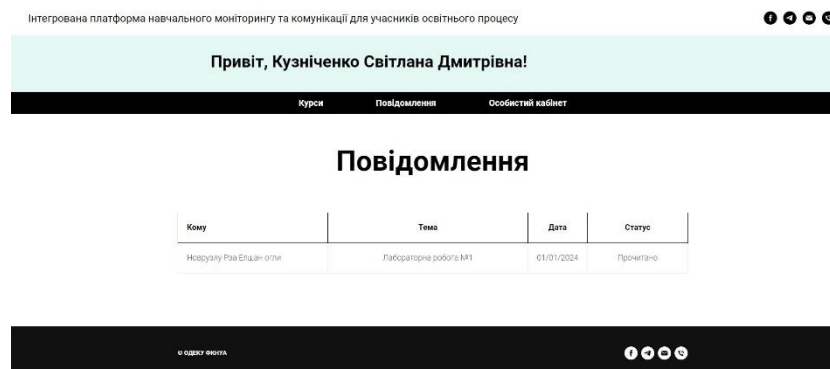


Рисунок 4.10 – Сторінка «Повідомлення» для користувача викладач

ВИСНОВКИ

У даній дипломній роботі було розроблено інтегровану платформу навчального моніторингу та комунікації для учасників освітнього процесу. Платформа складається з таких модулів: модуль авторизації та аутентифікації, модуль навчального моніторингу, модуль освітньої комунікації, модуль управління навчальним процесом, модуль інструментів для спільного навчання.

Інтегрована платформа навчального моніторингу та комунікації має ряд переваг: підвищує ефективність навчання, покращує співпрацю, знижує витрати, підвищує доступність.

Проведене тестування та налагодження платформи показало, що вона відповідає всім поставленим вимогам. Платформа зручна у використанні та має високу продуктивність.

Результати дипломної роботи можуть бути використані для вдосконалення освітнього процесу в навчальних закладах різного рівня. Інтегрована платформа навчального моніторингу та комунікації може стати цінним інструментом для викладачів та студентів.

Перспективами подальших досліджень є: розширення функціональності платформи, розробка мобільного додатку для платформи, впровадження платформи в навчальний процес конкретного навчального закладу.

Вважаю, що дана дипломна робота є актуальним та корисним дослідженням, яке може зробити значний внесок у розвиток освітніх технологій.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Connolly, T., & Begg, C. – Database Systems: A Practical Approach to Design, Implementation, and Management. – Pearson – 2014 – 6th Edition – 1440 сторінок – ISBN 978-0132943260;
2. Flanagan, D. – JavaScript: The Definitive Guide. – O'Reilly Media – 2020 – 7th Edition – 706 сторінок – ISBN 978-1491952023;
3. Fowler, M. – Patterns of Enterprise Application Architecture. – 2003 – 1st Edition. – Addison-Wesley Professional – 533 сторінок – ISBN 978-0321127426;
4. Hunt, A., & Thomas, D. – The Pragmatic Programmer: Your Journey to Mastery. – Addison-Wesley Professional – 1999 – 1st Edition – 352 сторінок – ISBN 978-0201616224;
5. Kroenke, D. M., Auer, D. J. – Database Concepts. – Pearson – 2013 – 6th Edition – 600 сторінок – ISBN 978-0132742924;
6. Pressman, R. S. – Software Engineering: A Practitioner's Approach. – McGraw-Hill Education – 2014 – 8th Edition – 976 сторінок – ISBN 978-0078022128;
7. Вовк, С. С. – Системний аналіз та проектування інформаційних систем. – Ліра-К – 2017 – 1-ше видання – 376 сторінок – ISBN 978-966-680-934-5;
8. Гергель, В. П., Мартиненко, О. В. – Системне програмування. – Ранок – 2019 – 1-ше видання – 448 сторінок – ISBN 978-966-641-270-5;
9. Завадський, І. О. – Інформаційні системи та технології. – Смолоскип – 2017 – 2-ге видання – 452 сторінок – ISBN 978-966-10-2677-0;
10. Колодяжний, А. М. – Бази даних: проектування, реалізація та супровід. – Центр учбової літератури – 2018 – 3-тє видання – 380 сторінок – ISBN 978-966-464-308-8;

11. Олійник, О. О. – Основи проектування інформаційних систем. – Магнолія-2006 – 2016 – 2-ге видання – 400 сторінок – ISBN 978-966-2478-92-7;
12. Столярчук, П. Г. – Веб-технології та веб-дизайн. – Знання – 2016 – 1-ше видання – 410 сторінок – ISBN 978-966-214-950-7;
13. Черненко, О. М. – Моделювання інформаційних систем. – Орієнтир – 2019 – 1-ше видання – 344 сторінки – ISBN 978-966-681-398-4.