МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»

ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ

КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

ПОЯСНЮВАЛЬНА ЗАПИСКА

до кваліфікаційної роботи освітнього ступеня «бакалавр»

за спеціальністю 121 «Інженерія програмного забезпечення»   
(освітня програма «Інженерія програмного забезпечення»)

на тему:  
«Система управління навчальним процесом для закладів освіти»

|  |
| --- |
| Виконав студент групи ІПЗ-20-2 БУБЕНКО Олексій Вікторович |
|  |
| Керівник роботи:  ВАКАЛЮК Тетяна Анатоліївна  Рецензент:  ВЛАСЕНКО Олег Васильович |
|  |
|  |

Житомир – 2024МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»

ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ

КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

**«ЗАТВЕРДЖУЮ»**

В.о. зав. кафедри інженерії програмного забезпечення

Андрій МОРОЗОВ

**ЗАВДАННЯ**

**на кваліфікаційну роботу**

Здобувач вищої освіти: **БУБЕНКО Олексій Вікторович**

Керівник роботи: **ВАКАЛЮК Тетяна Анатоліївна**

Тема роботи: **«**Система управління навчальним процесом для закладів освіти**»**,

затверджена Наказом закладу вищої освіти від

Вихідні дані для роботи:

Консультанти з бакалаврської кваліфікаційної роботи із зазначенням розділів,

що їх стосуються:

|  |  |  |  |
| --- | --- | --- | --- |
| **Розділ** | **Консультант** | Завдання видав | Завдання прийняв |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |

**Календарний план**

|  |  |  |  |
| --- | --- | --- | --- |
| **№**  **з/п** | **Назва етапів випускної кваліфікаційної роботи** | **Термін виконання етапів роботи** | **Примітка** |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 4 |  |  |  |
| 5 |  |  |  |
| 6 |  |  |  |
| 7 |  |  |  |
| 8 |  |  |  |
| 9 |  |  |  |
| 10 |  |  |  |

**РЕФЕРАТ**

Зм.

Арк.

№ докум.

Підпис

Дата

Арк.

2

ІПЗ.КР.Б – 121 – 24 – ПЗ

Розроб.

О. В. Бубенко

Керівник

.

Т. А. Вакалюк

Рецензент

О.В. Власенко

Зав. каф.

*А.В.Морозов*

**Система управління навчальним процесом для закладів освіти**

**Пояснювальна записка**

Літ.

Аркушів

71

Житомирська політехніка,

група ІПЗ-20-2

**ABSTRACT**

**ЗМІСТ**

[ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ 5](#_Toc159965992)

[ВСТУП 6](#_Toc159965993)

[Розділ 1. АНАЛІЗ НАПРЯМКІВ ВИКОРИСТАННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ДЛЯ РЕАЛІЗАЦІЇ СИСТЕМИ УПРАВЛІННЯ НАВЧАЛЬНИМ ПРОЦЕСОМ ДЛЯ ЗАКЛАДІВ ОСВІТИ 7](#_Toc159965994)

[1.1 Постановка задачі 7](#_Toc159965995)

[1.2 Аналіз аналогів програмного продукту 7](#_Toc159965996)

[1.3 Вибір архітектури системи управління навчальним процесом для закладів освіти 7](#_Toc159965997)

[1.4 Обґрунтування вибору інструментальних засобів та вимоги до апаратного забезпечення 7](#_Toc159965998)

[1.5 Висновки до розділу 1 7](#_Toc159965999)

**ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ**

**ВСТУП**

**Актуальність теми.** Сучасний світ переживає глобальні трансформації та зміни, які впливають на всі сфери життя, зокрема й на освіту. Сьогодні освітні заклади стикаються зі складністю навчального процесу, який вимагає постійної адаптації до нових умов та викликів. У цьому контексті система управління навчальним процесом стає головним інструментом, який дозволяє забезпечити ефективну та якісну освітню діяльність.

Система управління навчальним процесом закладу освіти — це комплекс програмних та технічних рішень, які допомагають організувати, контролювати, аналізувати навчальний процес в онлайн-режимі. Вона надає студентам та викладачам, учням та вчителям можливість отримувати та надавати освіту з будь-якого куточку світу, дозволяє зробити процес навчання більш індивідуальним, сприяє гнучкому навчанню, враховуючи потреби кожного. Також впровадження системи управління навчальним процесом може відкрити можливості для отримання освіти студентами та учнями, які перебувають на тимчасово окупованих територіях. Тому розробка такої системи є особливо актуальною в наш час.

**Метою** кваліфікаційної роботи

**Об’єктом дослідження** є

**Предметом дослідження** є

1. **АНАЛІЗ НАПРЯМКІВ ВИКОРИСТАННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ДЛЯ РЕАЛІЗАЦІЇ СИСТЕМИ УПРАВЛІННЯ НАВЧАЛЬНИМ ПРОЦЕСОМ ДЛЯ ЗАКЛАДІВ ОСВІТИ**

## Постановка задачі

Призначення системи полягає в інтеграції її в освітні заклади з метою забезпечення доступу до системи управління навчальним процесом через глобальну мережу Інтернет. Ця система дозволяє вчителям та викладачам створювати курси з різних навчальних дисциплін, завантажувати матеріали, тести та завдання для отримувачів освітніх послуг. Окрім цього, вона дозволяє вчителям оцінювати успішність учнів та студентів, виставляти оцінки за завдання, вести журнал успішності та надавати конструктивний зворотний зв'язок для поліпшення освітнього процесу.

Мета впровадження системи управління навчальним процесом для закладів освіти полягає у створенні максимально доступного, ефективного та інноваційного середовища для освіти, спрямованого на полегшення доступу до якісної освіти для отримувачів освіти у навчальних закладах через глобальну мережу Інтернет. Ця платформа має націлена сприяти викладачам та вчителям у створенні та розробці навчальних курсів, а учням та студентам надати можливість доступу до різноманітних матеріалів та інструментів навчання і відстежувати власну успішність. Все це, зі свого боку, сприятиме покращенню якості освіти та навчального процесу загалом.

Необхідна реалізація наступних кроків:

1. Аналіз функціональних вимог:
   * Провести аналіз потреб та очікувань різних груп користувачів системи управління навчальним процесом, таких як: студенти, учні викладачі, адміністрація тощо.
   * Визначити основні функції та можливості, які повинна мати система для забезпечення ефективності та якості навчання.
2. Вибір технічних інструментів:
   * Обрати найефективніший підхід для розробки власної системи, враховуючи специфіку та складність проекту, а також ресурси та обмеження.
   * Обрати оптимальний стек технологій для розробки веб-застосунку, визначити та обґрунтувати використання фреймворку ASP.NET Core для розробки серверної логіки та API, бібліотеки React для клієнтської частини та базу даних PostgreSQL для зберігання даних, а також інші інструменти та бібліотеки, які будуть необхідні для реалізації функціональних вимог.
3. Розробка концепції та архітектури системи:
   * Розробити концептуальну модель системи, використовуючи діаграми UML, такі як діаграми прецедентів, класів, послідовності, активності тощо.
   * Розробити архітектуру системи, використовуючи шаблони проєктування, принципи та стандарти, такі як: MVC, REST, SOLID, DRY тощо.
4. Реалізація серверної та клієнтської частин системи:
   * Реалізувати серверну частину додатку на базі фреймворку ASP.NET Core та Entity Framework Core для забезпечення надійного та швидкого API, використовуючи базу даних PostgreSQL для ефективного зберігання даних про різноманітний навчальний контент, користувачів тощо.
   * Реалізувати клієнтську частину додатку на базі бібліотек React, NextJS та Tailwind для створення зручного та реактивного інтерфейсу, забезпечити користувачам можливість легкої навігації, пошуку та доступу до навчальних матеріалів, тестів та завдань тощо.
5. Реалізація функцій оцінювання та аналізу успішності:
   * Реалізувати функції оцінювання успішності учнів та студентів, засновані на аналізі їхніх результатів, забезпечити викладачам можливість виставляти оцінки за завдання, вести журнал успішності та надавати конструктивний зворотний зв’язок для поліпшення навчального процесу.
   * Реалізувати функції аналітики та звітності, що дозволяють відстежувати та візуалізувати різноманітну статистику та динаміку навчання, використовуючи різні такі інструменти як: графіки, діаграми, таблиці тощо.
6. Тестування та оптимізація продукту:
   * Виконати мануальні тестування всіх функціональних компонентів. Реалізувати юніт-тести для окремих функціональних компонентів, щоб усунути помилки на ранніх етапах розробки та забезпечити стабільність і надійність програмного забезпечення.
   * Вдосконалення продуктивності та ефективності додатку досягається за допомогою оптимізації, що включає в себе використання кешування, мініфікацію та інші методи.

## Аналіз аналогів програмного продукту

На сьогодні в освітній сфері системи управління навчальним процесом виступають основним інструментом для забезпечення ефективного функціонування освітніх закладів. З метою створення багатофункціональної та конкурентоздатної системи управління, що відповідає викликам сучасної освіти, важливим етапом є порівняння існуючих аналогів.

Аналіз подібних систем дозволить визначити основні функціональні можливості. Детальне порівняння аналогів управління навчальним процесом допоможе виділити важливі аспекти, такі як адміністрування, створення навчальних матеріалів, перегляд аналітичної інформації тощо. Ці аспекти надалі визначатимуть успішність та конкурентоздатність нашого програмного продукту.

«**HUMAN Школа»** — система управління навчанням, що об'єднує всіх учасників освітнього процесу та допомагає вчителям та керівникам шкіл організувати внутрішні бізнес-процеси закладу освіти. Система HUMAN надає розширені можливості для централізації освітніх процесів у навчальному закладі. Зокрема, можливість ведення всієї документації в електронному форматі і зосередження її на єдиній платформі спрощує рутинні завдання. Варто відзначити доступ до архіву матеріалів, наказів, та іншої інформації, що сприяє зосередженню всіх необхідних даних на одній платформі.

Однією з ключових особливостей цього рішення є можливість створення власної шкільної соцмережі. Адміністратори та викладачі можуть здійснювати ефективну комунікацію шляхом поширення новин, оголошень, створення подій та проведення опитувань, надаючи учасникам освітнього процесу актуальну інформацію.

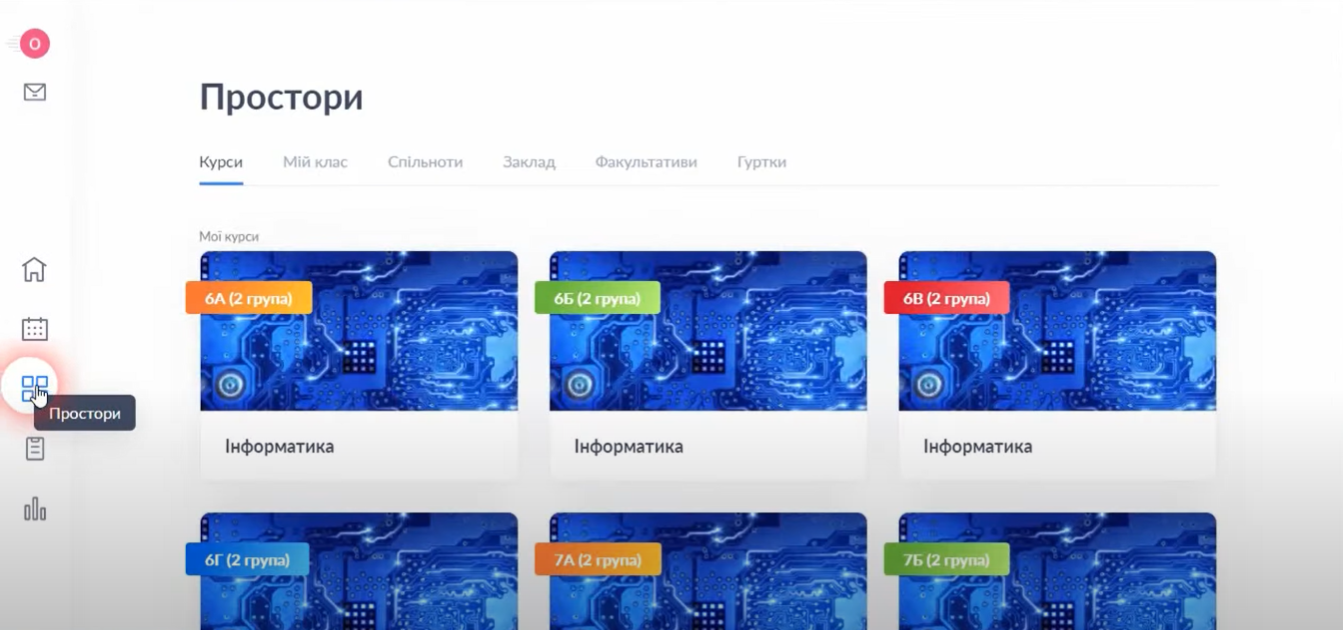
Система HUMAN також відзначається можливістю моніторингу та аналізу якості освіти в закладі. Адміністратор отримує доступ до інформації про діяльність педагогічних працівників, дозволяючи переглядати навчальні плани, матеріали та домашні завдання у режимі реального часу. Завдяки аналітиці системи, адміністратор навчального закладу має змогу відстежити різноманітні показники та аспекти функціонування заклад, такі як: успішність та відвідуваність учнів, ефективність роботи вчителів та викладачів. Це забезпечує високий рівень контролю освітнього процесу і дозволяє своєчасно впливати на його якість.

Переваги:

* Централізованість всіх необхідних інструментів для управління освітнім процесом.
* Гнучкість для дистанційного або змішаного навчання з різними форматами.
* Поглиблена аналітика про успішність та відвідуваність учнів.
* Інструменти для рефлексії учнів для персоналізованого навчання та розвитку талантів.
* Універсальний розклад для планування та проведення занять.
* Адміністрування системи для створення та керування класами, курсами, користувачами тощо.
* Підтримка та консультація від команди HUMAN для адаптації шкіл до сучасних освітніх інновацій.

Недоліки:

* Дуже обмежена підтримка типів файлів, які можна завантажувати та переглядати на платформі.
* Відсутність сповіщень про нові завдання або коментарі від вчителів та учнів.
* Відсутність дати перевірки домашнього завдання вчителем.
* Відсутність функціональності, яка дозволяє переглянути чи прочитане повідомлення у внутрішній соціальній мережі.



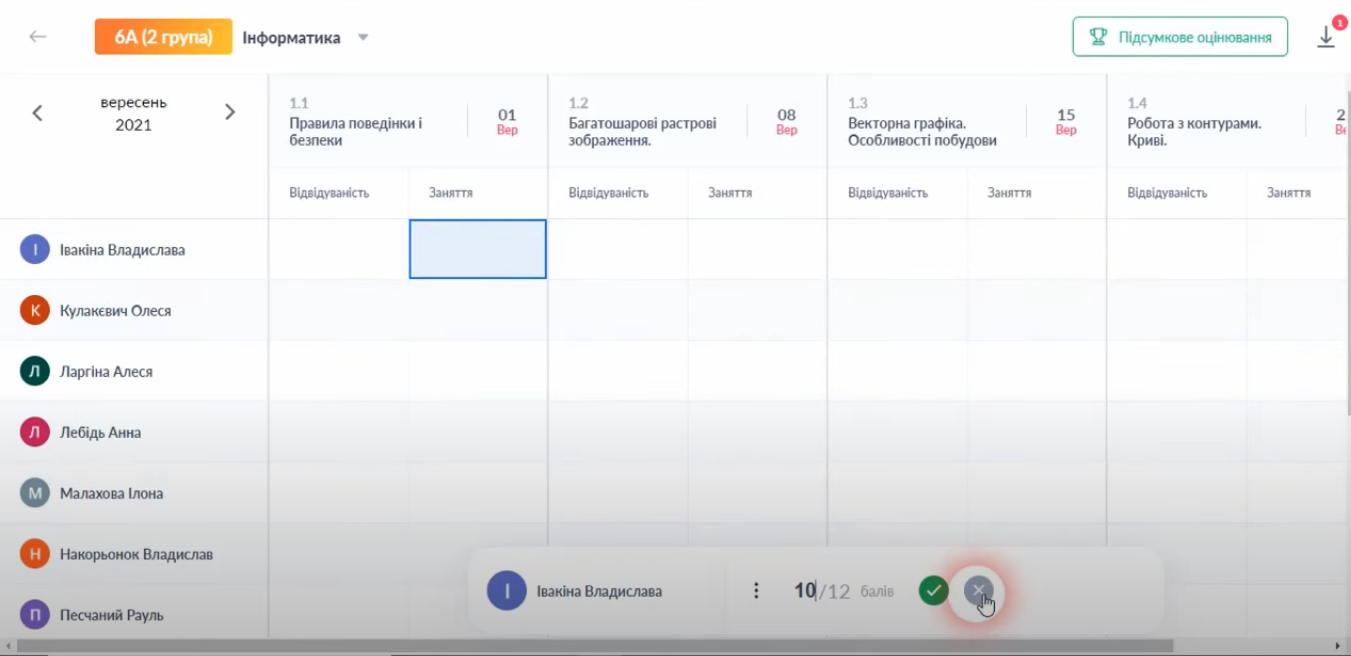


Рис. 1.2.1 – 1.2.2. Інтерфейс «HUMAN Школа»

«**Нові знання»** — платформа для електронних щоденників та журналів з інструментами для дистанційного навчання. Надає можливості створення уроків, виставлення оцінок та аналізу успішності учнів, класів, школи.

Платформа «Нові знання» дозволяє впроваджувати щоденники, які надають учням та їхнім батькам постійний доступ до історії отриманих оцінок та домашніх завдань. Ця функція дозволяє проводити аналіз успішності за тривалим періодом.

Також платформа «Нові знання» оснащена розгалуженими звітами, які забезпечують гнучкі механізми аналізу та графічного відображення навчальних досягнень учнів, класу, школи та роботи вчителя. Додатково, платформа реалізує можливості дистанційного навчання, дозволяючи виконувати, перевіряти та обговорювати практичні завдання онлайн.

Переваги:

* Є можливість завантажувати власні завдання, користуватися електронними підручниками та бібліотекою уроків.
* Автоматичне розрахування та формування звітів для всіх видів оцінок, включаючи тематичні, семестрові, річні оцінки.
* Є можливість роздрукувати журнал або окрему сторінку предмета.

Недоліки:

* Сервери можуть бути недоступними при великій кількості одночасно працюючих користувачів.
* Певні елементи інтерфейсу є застарілими.

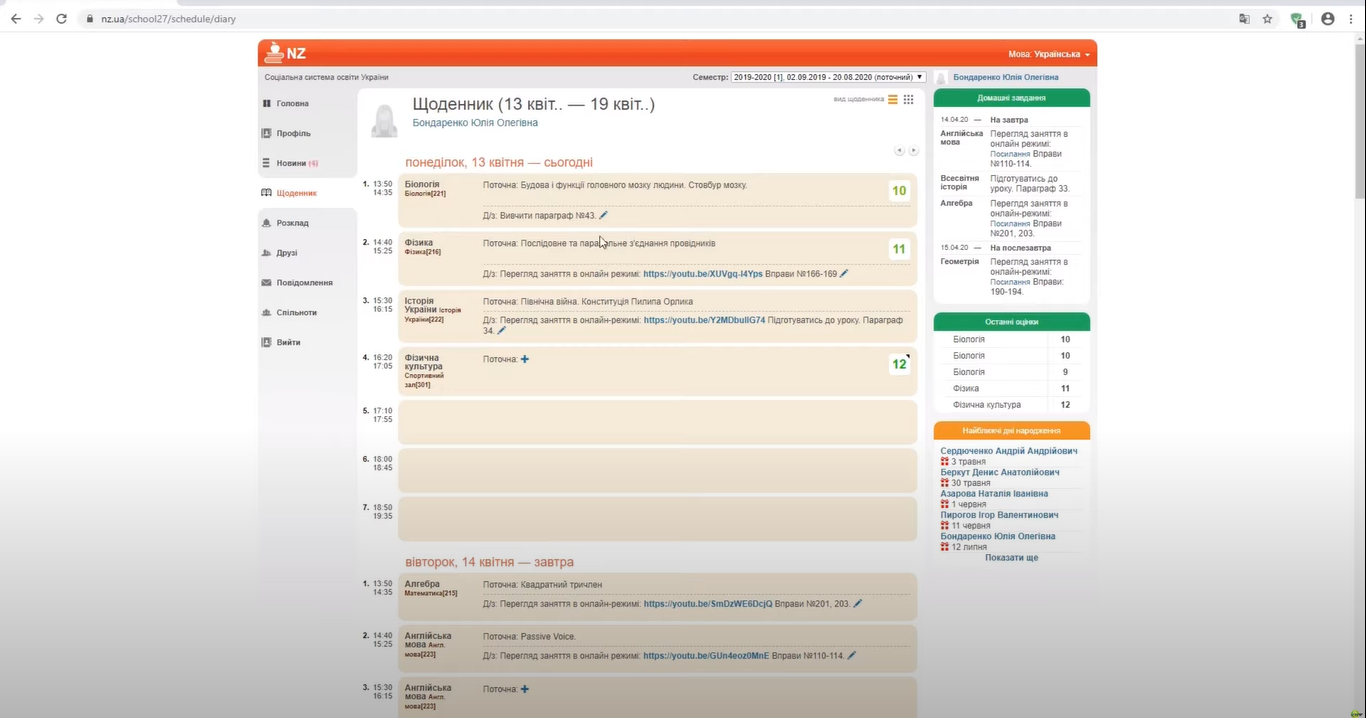


Рис. 1.2.3. Інтерфейс «Нові знання»

«**Smart school»** — це система автоматизації освітнього процесу для закладів загальної середньої освіти, професійно-технічних навчальних закладів та вищих навчальних закладів I - II рівнів акредитації.

«Smart School» — це комплексна освітня платформа, яка спрямована на оптимізацію адміністративних процесів, покращення комунікації між учасниками освітнього процесу та забезпечення безперервного навчання. Платформа надає можливості для адміністраторів, викладачів, учнів та батьків.

Адміністратори можуть керувати розкладом занять, мають доступ до журналів успішності всіх класів, можуть переглядати звітність, керувати списками учнів, батьків та викладачів, тарифікувати викладачів та розподіляти робочий час.

Викладачі можуть працювати з журналами, зв'язуватися з батьками, розміщувати навчальні матеріали, створювати власні навчальні плани, керувати класом та створювати онлайн-тести.

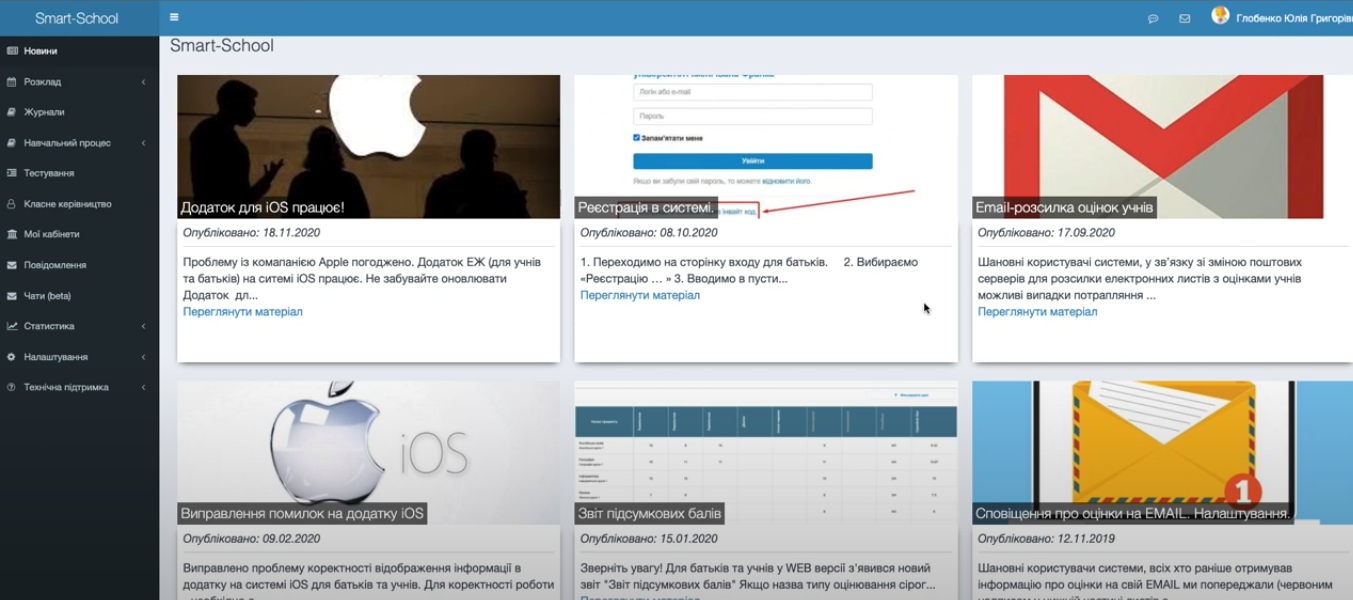
Учні та їх батьки можуть отримувати актуальну інформацію про оцінки, відвідування занять, розклад занять та його зміни, домашні завдання, графіки успішності, зв'язок з вчителями та проходити онлайн-тестування.

Переваги:

* Електронний журнал дозволяє автоматично генерувати звіти про успішність та зручно вести облік відвідування занять.
* Редактор розкладу занять дозволяє легко формувати графік проведення занять та легко вносити зміни до нього.
* Можливість формувати різноманітні типи звітів для навчальних закладів, що спрощує процес аналізу та моніторингу навчальних досягнень.

Недоліки:

* Перевантажений інтерфейс таких сторінок, як журнал оцінок
* Різні посилання для входу на платформу для учнів та батьків і вчителів



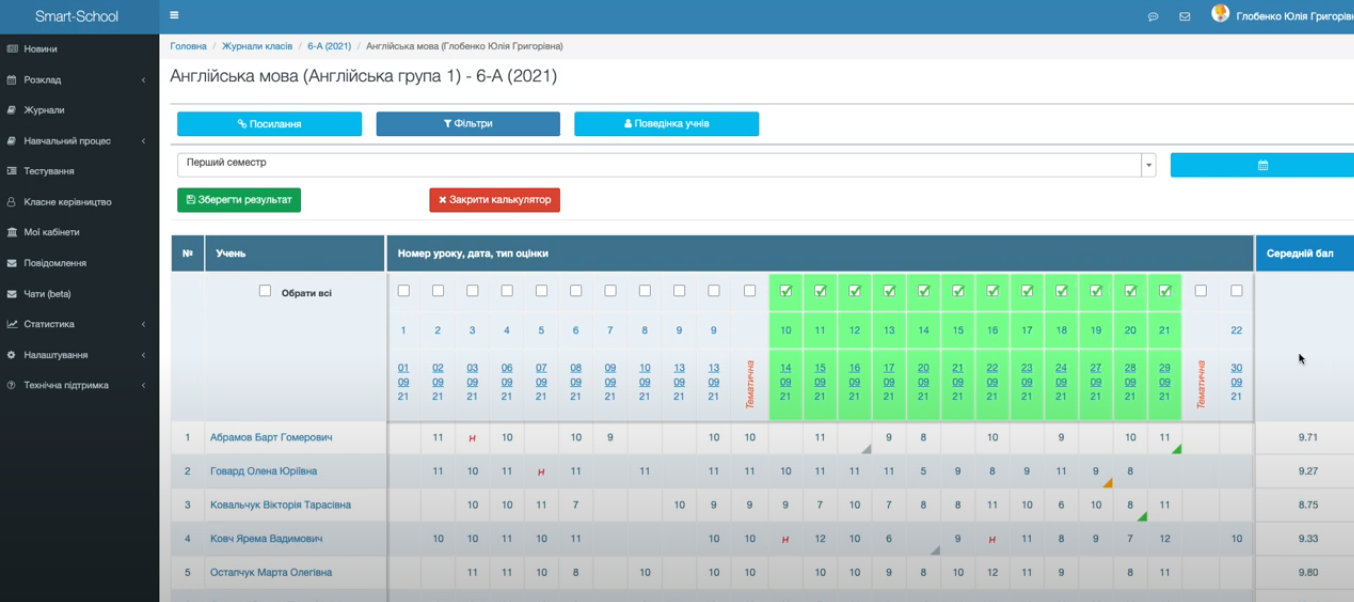
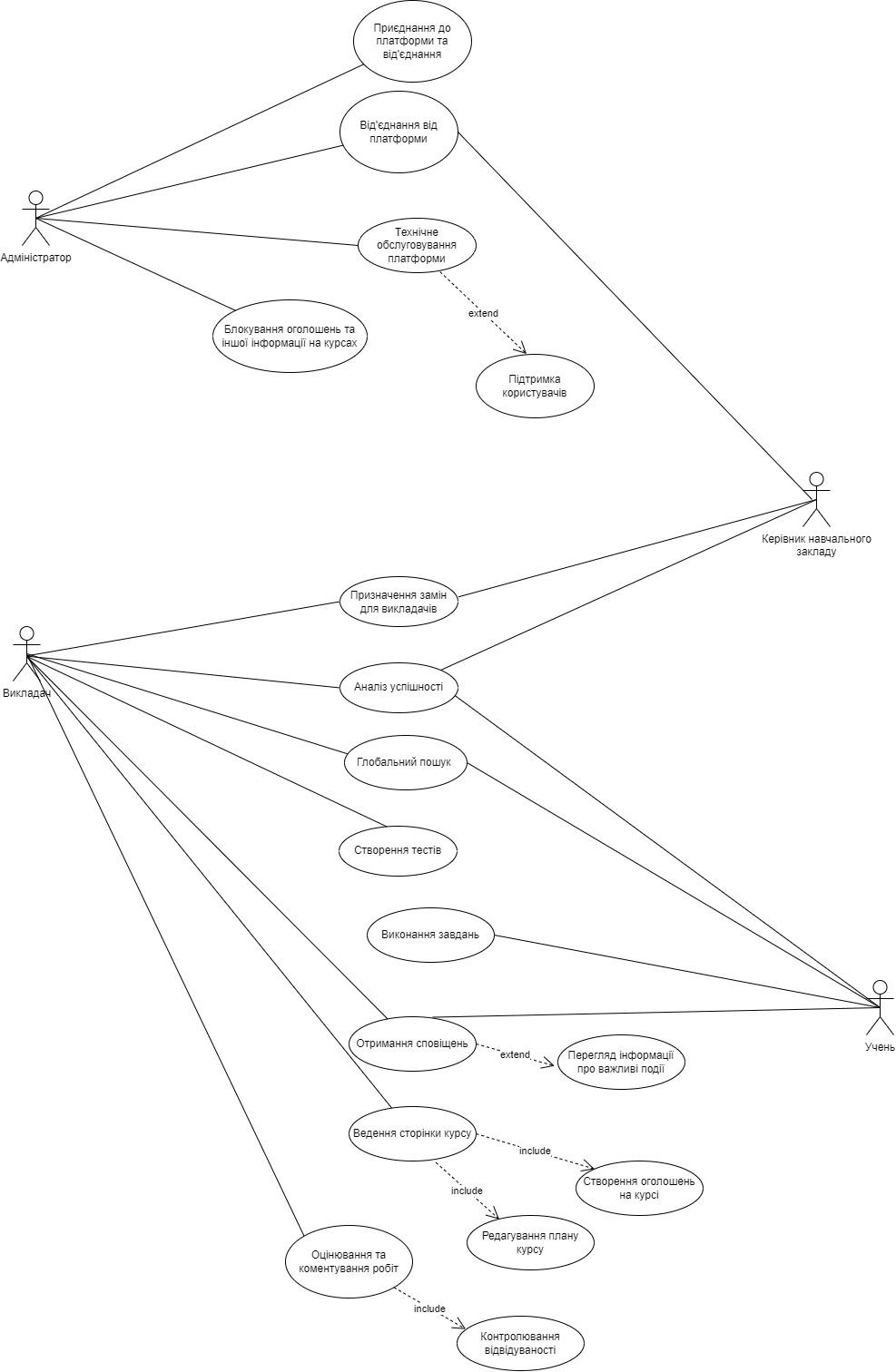


Рис. 1.2.4. – 1.2.5. Інтерфейс «Smart school»

Таблиця 2.1. Порівняння аналогів

|  |  |  |  |
| --- | --- | --- | --- |
| **Характеристика** | **«Human Школа»** | **«Нові знання»** | **«Smart school»** |
| Електронний журнал | + | + | + |
| Комунікація учнів та вчителів | + | - | + |
| Автоматична звітність | + | + | + |
| Система сповіщень | ± | - | + |
| Відстеження прогресу учнів | + | + | + |
| Можливості тестування учнів | + | - | + |
| Завантаження власного контенту вчителем на сторінку курсу | + | - | + |

Після аналізу освітніх платформ «Human Школа», «Нові знання» та «Smart School» вдалось виявити кілька основних характеристик для врахування при розробці нашого веб-застосунку. Отже, головні функції, на які варто звернути увагу, включають керування учнями, вчителями, навчальними курсами тощо, аналіз успішності учнів та формування відповідної звітності, систему сповіщень, ведення сторінки курсу — наповнення її теоретичним контентом, практичними завданнями та тестами.



## Вибір архітектури системи управління навчальним процесом для закладів освіти

Вибір архітектури для системи управління навчальним процесом є важливим етапом, який впливає на успіх проєкту. Архітектура визначає структуру системи, включаючи способи взаємодії між різними компонентами. Вона впливає на багато аспектів системи, включаючи її продуктивність, надійність, масштабованість, безпеку тощо. Вибір архітектури — це рішення, яке впливає на довгострокову ефективність та гнучкість системи.

Розглянемо три основні архітектурні підходи: монолітний, мікросервісний та сервісно-орієнтований. Кожен з них має свої особливості, переваги та недоліки, які ми проаналізуємо детальніше.

Монолітна архітектура — це традиційний архітектурний підхід, де весь функціонал розташований в одному місці. У цій архітектурі всі компоненти застосунку спільно працюють із однією базою даних та іншими ресурсами, що робить розробку та розгортання простішими, але може призвести до складнощів з масштабованістю та підтримкою доступності для великої кількості користувачів.

Мікросервісна архітектура – це архітектурний підхід, коли система розбивається на невеликі, незалежні сервіси, які можуть бути розгорнуті окремо. Кожен сервіс відповідає за певний функціонал та має власну базу даних. Ця архітектура забезпечує гнучкість, масштабованість та можливість швидкого реагування на зміни, але приносить додаткову складність у керуванні та взаємодії між сервісами.

Сервісно-орієнтована архітектура — це архітектурний підхід, коли функціональність розбивається на незалежні сервіси, які виконують конкретні функції та можуть бути використані в різних контекстах. Ця архітектура сприяє повторному використанню коду, забезпечує гнучкість та складається з сервісів, які можуть бути використані через систему керування пакунками. Сервісно-орієнтована архітектура може стати причиною складнощів у конфігурації, відлагодженні та керуванні залежностями між сервісами, що потребує додаткових зусиль для забезпечення ефективного функціонування системи.

Вибір мікросервісної архітектури для системи управління навчальним процесом обґрунтований кількома важливими факторами. Перш за все, ця архітектура забезпечує гнучкість і розширюваність, дозволяючи легко адаптувати систему до зростаючих потреб системи. Кожен логічний компонент може бути розглянутий як окремий сервіс, що дозволяє незалежно розвивати та масштабувати його.

Другим важливим аргументом є ефективне управління та підтримка системи. У мікросервісній архітектурі окремі сервіси можуть бути розроблені, відтестовані та підтримувані окремо, що спрощує процес розробки та забезпечує швидке впровадження змін. Крім того, ця архітектура забезпечує більшу стабільність, оскільки проблеми в одному сервісі не впливають на решту системи. Такий підхід полегшує виявлення та виправлення помилок, що виникають у процесі роботи системи.

Отже, компоненти веб-орієнтованої мікросервісної архітектури включають перелік незалежних сервісів, які співпрацюють для забезпечення повноцінного функціоналу системи. Кожен з цих сервісів спрямований на виконання конкретної функції та може бути розгорнутий і масштабований окремо від інших. Серед найважливіших компонентів можна виділити сервіси автентифікації та авторизації, які відповідають за перевірку ідентифікації користувачів та надання доступу до ресурсів системи.

Додатковим важливим компонентом веб-орієнтованої мікросервісної архітектури є API Gateway. Це сервіс, який виступає в якості центральної точки входу для всіх запитів до систем. Основні функції API Gateway включають в себе маршрутизацію запитів до відповідних мікросервісів, забезпечення автентифікації та авторизації клієнтів, перетворення та агрегацію даних та логування запитів.

Кожен сервіс має власну базу даних, що дозволяє кожному сервісу мати свою власну структуру даних та зручність в керуванні ними. Це сприяє зменшенню залежностей між сервісами. Кожен сервіс може використовувати технології та схеми баз даних, які найкраще відповідають його потребам, що забезпечує оптимальну продуктивність та ефективність. Тож такий підхід дозволяє кожному сервісу бути гнучким та адаптуватися до власних конкретних вимог.

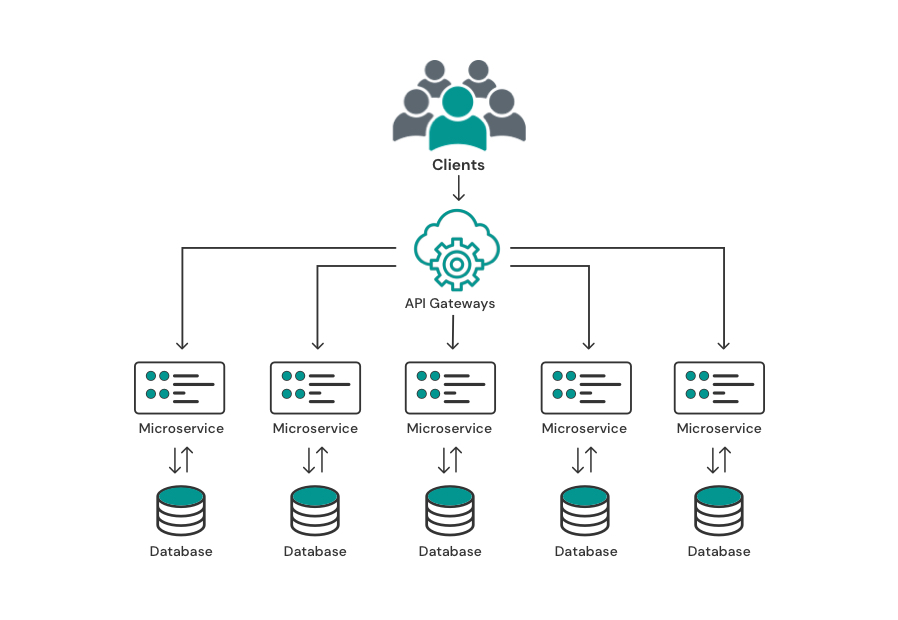


Рис. 1.3.1. Мікросервісна архітектура

Отже, на серверній частині системи управління навчальним процесом планується впровадження мікросервісної архітектури. Для розробки мікросервісів та АРІ та буде використано фреймворк ASP .NET Core та технологію MassTransit для забезпечення комунікації між цими сервісами. Кожен з мікросервісів буде відповідати за певну функціональність, таку як управління користувачами, організація курсів, оцінювання студентів, формування звітів тощо.

На клієнтській частині системи будуть використані бібліотеки React та …

## Обґрунтування вибору інструментальних засобів та вимоги до апаратного забезпечення

## Висновки до розділу 1