

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

«Веборієнтована навчальна платформа гімназії»

Студентки 2мз курсу, 2
групи,
спеціальності 121 «Інженерія
програмного забезпечення»
освітньої програми «Інженерія
програмного забезпечення»

Більської Анастасії
Русланівни

підпис студента

Науковий керівник
кандидат педагогічних наук,
доцент кафедри інженерії
програмного забезпечення та
кібербезпеки

Котенко Наталія
Олексіївна

підпис керівника

Гарант освітньої програми
кандидат педагогічних наук,
доцент кафедри інженерії
програмного забезпечення та
кібербезпеки

Котенко Наталія
Олексіївна

підпис гаранта

Факультет інформаційних технологій

Кафедра інженерії програмного забезпечення та кібербезпеки

Освітній ступінь магістр

Освітня програма 121 «Інженерія програмного забезпечення»

Затверджую

Зав. кафедри інженерії програмного
забезпечення та кібербезпеки

Криворучко О. В.

«13» грудня 2022 р.

Завдання

на випускн кваліфікаційну роботу студентів

Більській Анастасії Русланівні

(прізвище, ім'я, по батькові)

1. Тема випускної кваліфікаційної роботи «Веборієнтована навчальна
платформа гімназії»

Затверджена наказом ректора від «09» грудня 2022 р. № 3339

2. Строк здачі студентом закінченої роботи 1 грудня 2023

3. Цільова установка та вихідні дані до роботи

Мета роботи розробка веборієнтованої навчальної платформи для гімназії з
ціллю підвищення ефективності навчання та розширення можливостей
освітнього процесу та розвитку цифровізації української освіти загалом.

Об'єкт дослідження вебтехнології в освіті.

Предмет дослідження розробка веборієнтованої навчальної платформи
гімназії.

4. Консультанти роботи із зазначенням розділів, які консультують:

Розділ	Консультант (прізвище, ініціали)	Підпис, дата	
		Завдання видав	Завдання прийняв

5. Зміст випускної кваліфікаційної роботи (перелік питань за кожним розділом)
ВСТУП

РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1. Основні поняття веборієнтованої платформи

1.2. Порівняльний аналіз вебплатформ та комп'ютерних та мобільних застосунків

1.3. Цифровізація в освіті

1.4. Висновки до розділу 1

РОЗДІЛ 2 ТЕХНОЛОГІЇ ТА ІНСТРУМЕНТИ ДЛЯ РОЗРОБКИ

ВЕБОРІЄНТОВАНОЇ НАВЧАЛЬНОЇ ПЛАТФОРМИ ГІМНАЗІЇ

2.1. Аналіз та підбір Frontend технологій

2.1.1. Способи реалізації Frontend

2.1.2. HTML та CSS

2.1.3. Порівняльний аналіз JavaScript та TypeScript

2.1.4. Порівняльний аналіз Frontend фреймворків. React

2.2. Аналіз та підбір Backend технологій. JavaScript, Node.js, Express.js

2.3. Аналіз та підбір системи управління базами даних

2.4. Висновки до розділу 2

РОЗДІЛ 3 РОЗРОБКА АРХІТЕКТУРИ ВЕБОРІЄНТОВАНОЇ НАВЧАЛЬНОЇ ПЛАТФОРМИ

3.1. Інформаційна система веборієнтованої навчальної платформи

3.1.1. Предметна область та концепція

3.1.2. Нефункціональні вимоги до веборієнтованої навчальної платформи

3.1.3. Проектна організація та опис бази даних

3.1.4. Дійові особи інформаційної системи

3.1.5. Опис функціональних вимог

3.2. Розробка дизайну

3.3. Backend веборієнтованої навчальної платформи

3.4. Frontend веборієнтованої навчальної платформи

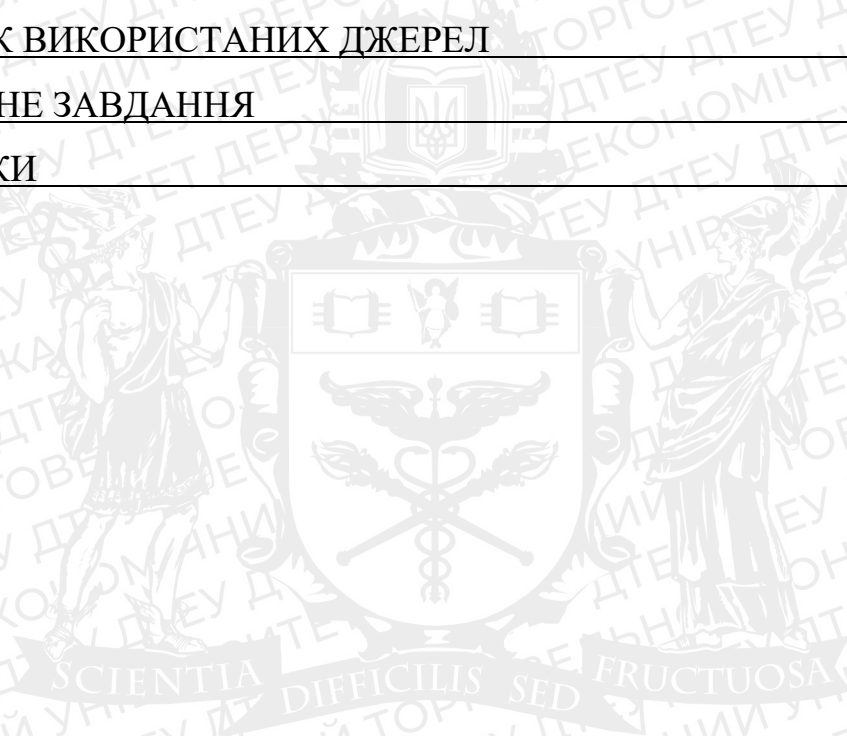
3.5. Висновок до розділу 3

ВИСНОВКИ ТА ПРОПОЗИЦІЇ

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

ТЕХНІЧНЕ ЗАВДАННЯ

ДОДАТКИ



6. Календарний план виконання роботи

№ пор.	Назва етапів випускної кваліфікаційної роботи	Строк виконання етапів роботи	
		за планом	фактично
1	2	3	4
1.	<i>Вибір теми випускної кваліфікаційної роботи</i>	07.11.2022	07.11.2022
2.	<i>Розробка та затвердження завдання на роботу магістра (стац/заоч)</i>	13.12.2022	13.12.2022
3.	<i>Вступ та перелік літературних джерел</i>	24.02.2023	24.02.2023
4.	<i>Розробка технічного завдання</i>	15.03.2023	15.03.2023
5.	<i>Розділ 1. Аналіз предметної області</i>	10.04.2023	10.04.2023
6.	<i>Розділ 2. Технології та інструменти для розробки веборієнтованої навчальної платформи гімназії</i>	24.05.2023	24.05.2023
7.	<i>Розділ 3. Розробка архітектури веборієнтованої навчальної платформи</i>	06.09.2023	06.09.2023
8.	<i>Розробка програми та методики тестування</i>	18.10.2023	18.10.2023
9.	<i>Написання наукової статті</i>	17.05.2023	17.05.2023
10.	<i>Керівництво користувача</i>	25.10.2023	25.10.2023
11.	<i>Висновки та пропозиції</i>	01.11.2023	01.11.2023
12.	<i>Здача випускної кваліфікаційної роботи на кафедрі (перша перевірка)</i>	06.11.2023	06.11.2023
13.	<i>Підготовка автореферату та презентації доповіді</i>	06.11.2023	06.11.2023
14.	<i>Попередній захист випускної кваліфікаційної роботи</i>	20.11.2023 – 24.11.2023	20.11.2023 – 24.11.2023
15.	<i>Здача зброшурованої випускної кваліфікаційної роботи</i>	1.12.2023	1.12.2023
16.	<i>Зовнішнє рецензування випускної кваліфікаційної роботи</i>	2.12.2023	2.12.2023
17.	<i>Підготовка до публічного захисту випускної кваліфікаційної роботи</i>	05.12.2023- 06.12.2023	05.12.2023- 06.12.2023

7. Дата видачі завдання «13» грудня 2022 р.

8. Науковий керівник випускної кваліфікаційної роботи _____

Котенко Н.О.

(прізвище, ініціали, підпис)

9. Гарант освітньої програми _____

Котенко Н.О.

(прізвище, ініціали, підпис)

10. Завдання прийняв до виконання студент _____

Більська А.Р.

(прізвище, ініціали, підпис)

АНОТАЦІЯ

Згідно із метою дослідження, робота присвячена розробці архітектури веборієнтованої навчальної платформи для гімназії, з ціллю підвищення ефективності навчання та розширення можливостей освітнього процесу й розвитку цифровізації української освіти загалом.

Випускна кваліфікаційна робота на тему «Веборієнтована навчальна платформа гімназії» містить 55 сторінок, 13 рисунків та 2 таблиці. Перелік використаних джерел налічує 22 найменувань.

Випускна кваліфікаційна робота містить три основні розділи.

У першому розділі було дано визначення таким поняттям як веборієнтована навчальна платформа, визначено переваги веборієнтованої навчальної платформи перед десктоп- та мобільним застосунком, розглянуто питання цифровізації освіти, необхідності наявності власної веборієнтованої навчальної платформи для закладів освіти, а також виконано порівняльний аналіз загальнодоступних веборієнтованих навчальних платформ.

У другому розділі було обрано основні технології для розробки веборієнтованої навчальної платформи, а саме технології для реалізації Frontend та Backend, а також систему управління базами даних.

У третьому розділі було спроектовано інформаційну систему, розроблено базу даних, архітектуру backend та frontend, а також розроблено дизайн для гілки перегляду та створення домашніх завдань веборієнтованої навчальної платформи.

Результатом роботи є створена архітектура веборієнтованої навчальної платформи.

Ключові слова: ВЕБОРІЄНТОВАНА НАВЧАЛЬНА ПЛАТФОРМА, ЦИФРОВІЗАЦІЯ ОСВІТИ, ДИСТАНЦІЙНА ОСВІТА, ТЕХНОЛОГІЇ ОСВІТИ.

ABSTRACT

According to the purpose of the study, the work was devoted to the development of the architecture of a web-oriented educational platform for a gymnasium, with the aim of increasing the effectiveness of education and expanding the possibilities of the educational process and the development of digitalization of Ukrainian education in general.

Graduation qualification work on the topic "Web-oriented educational platform of the gymnasium" contains 55 pages, 13 figures and 2 tables. The list of used sources includes 22 items.

The graduation thesis contains three main sections.

In the first section, the concept of a web-based educational platform was defined, the advantages of a web-based educational platform over a desktop and mobile application were determined, the issue of digitalization of education was considered, the need for an own web-based educational platform for educational institutions, and a comparative analysis of publicly available web-based educational platforms was also performed.

In the second section, the main technologies for the development of a web-oriented educational platform were selected, namely technologies for the implementation of Frontend and Backend, as well as a database management system.

In the third section, the information system was designed, the database, backend and frontend architecture were developed, and the design for the review branch and homework creation of the web-oriented educational platform was developed.

The result of the work is the created architecture of a web-oriented educational platform.

Keywords: WEB-ORIENTED EDUCATIONAL PLATFORM, DIGITALIZATION OF EDUCATION, DISTANCE EDUCATION, EDUCATION TECHNOLOGIES.

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

CRUD: Create, Read, Update, Delete – операції для управління даними в базі даних.

SEO: Search Engine Optimization – оптимізація для покращення видимості сайту в пошукових системах.

UX/UI: User Experience/User Interface – дизайн та користувацький досвід вебсайту.

SPA: Single Page Application – вебзастосунок, що працює на одній сторінці без перезавантаження.

DOM: Document Object Model – структура та представлення вебсторінки для програмної обробки.

LMS (англ. Learning Management System) – система управління навчанням.

SCORM (Sharable Content Object Reference Model) – набір стандартів, що використовується у системах дистанційного навчання.

Mobile Learning (M-Learning) – навчання, доступне на мобільних пристроях.

Run-Time Environment (RTE) – середовище, для взаємодії з навчальними матеріалами.

Sharable Content Objects (SCO) – найменші навчальні одиниці, які можуть бути розроблені, оцінені та використані незалежно від інших.

<i>ДТЕУ 121 023-02.МР</i>				
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>
Зав. каф.		Криворучко О.В.		19.09.23
Керівник		Котенко Н.О.		19.09.23
Гарант		Котенко Н.О.		19.09.23
Розробив		Більська А.Р.		19.09.23
<i>Веборієнтована навчальна платформа гімназії</i>				
<i>Перелік умовних скорочень</i>				
<i>Стадія</i>		<i>Аркуш</i>	<i>Аркушів</i>	
<i>ПС</i>		<i>2</i>	<i>55</i>	
<i>Факультет інформаційних технологій</i>				
<i>2мз курс, 2 група</i>				

ЗМІСТ

ВСТУП.....	4
РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	8
1.1. Основні поняття веборієнтованої платформи	8
1.2. Порівняльний аналіз вебплатформ, комп'ютерних та мобільних застосунків	8
1.3. Цифровізація в освіті.....	10
1.4. Висновок до розділу 1	14
РОЗДІЛ 2 ТЕХНОЛОГІЇ ТА ІНСТРУМЕНТИ ДЛЯ РОЗРОБКИ ВЕБОРІЄНТОВАНОЇ НАВЧАЛЬНОЇ ПЛАТФОРМИ ГІМНАЗІЇ.....	16
2.1. Аналіз та підбір Frontend технологій.....	16
2.1.1. Способи реалізації Frontend.....	16
2.1.2. HTML та CSS	18
2.1.3. Порівняльний аналіз JavaScript та TypeScript.....	20
2.1.4. Порівняльний аналіз Frontend фреймворків. React	21
2.2. Аналіз та підбір Backend технологій. JavaScript, Node.js, Express.js.....	23
2.3. Аналіз та підбір системи управління базами даних.....	26
2.4. Висновок до розділу 2	29
РОЗДІЛ 3 РОЗРОБКА АРХІТЕКТУРИ ВЕБОРІЄНТОВАНОЇ НАВЧАЛЬНОЇ ПЛАТФОРМИ.....	32
3.1. Інформаційна система веборієнтованої навчальної платформи	32
3.1.1. Предметна область та концепція.....	32
3.1.2. Нефункціональні вимоги до веборієнтованої навчальної платформи	33
3.1.3. Проектна організація та опис бази даних.....	34
3.1.4. Дійові особи інформаційної системи.....	37
3.1.5. Опис функціональних вимог	37
3.2. Розробка дизайну	39
3.3. Backend веборієнтованої навчальної платформи	43
3.4. Frontend веборієнтованої навчальної платформи.....	44
3.5. Висновок до розділу 3	47
ВИСНОВКИ ТА ПРОПОЗИЦІЇ.....	50
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	53
ТЕХНІЧНЕ ЗАВДАННЯ	56
ДОДАТКИ.....	58

<i>ДТЕУ 121 023-02.МР</i>				
Зм.	Аркуш	№ докум.	Підпис	Дата
Зав. каф.		Криворучко О.В.		01.11.23
Керівник		Котенко Н.О.		01.11.23
Гарант		Котенко Н.О.		01.11.23
Розробив		Більська А.Р.		01.11.23
<i>Веборієнтована навчальна платформа гімназії</i>				
<i>Зміст</i>				
		<i>Факультет інформаційних технологій</i>		
		<i>2мз курс, 2 група</i>		
		<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
		<i>Зміст</i>	3	55

ВСТУП

Актуальність. У сучасному світі, де технології швидко розвиваються, інформаційне середовище стає невід'ємною складовою освіти. У школах і гімназіях стає все більш актуальною необхідність впровадження нових технологій для покращення процесу навчання та сприяння залученню учнів до активної освітньої діяльності. Веборієнтовані навчальні платформи стають потужним інструментом, який може сприяти модернізації освітнього процесу та підвищенню якості навчання.

Пандемія COVID-19 впровадила стрімке використання цифрових технологій в освіті та стрімко діджиталізувало українські школи. В умовах військової агресії росії проти України, здобувачі середньої та початкової освіти вимушені періодично переходити на дистанційний режим навчання у зв'язку із заходами безпеки, що ще раз підкріплює важливість наявності якісної веобрієнтованої платформи, яка б мала не лише освітній функціонал, а й інформативний, що дозволило б ознайомити батьків дітей із гімназією, та повідомити про наявні та перенесені події в гімназії тощо.

Крім цього, державні програми розроблені з метою цифровізації освіти ще раз показують, що це той вектор, в якому необхідно рухатись з метою розвитку вітчизняного освітнього процесу з метою покращення та оновлення процесу отримання знань для здобувачів освіти та полегшення роботи викладачів.

Наукова новизна дослідження. Наукова новизна дослідження полягає в розробці інноваційної веборієнтованої навчальної платформи, яка відповідає сучасним тенденціям у галузі освіти та забезпечує інтерактивність,

					<i>ДТЕУ 121 023-02.МР</i>			
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>	<i>Веборієнтована навчальна платформа гімназії</i>	<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
Зав. каф.	Криворучко О.В.			24.02.23		В	4	55
Керівник	Котенко Н.О.			24.02.23		<i>Факультет інформаційних технологій</i>		
Гарант	Котенко Н.О.			24.02.23		<i>2мз курс, 2 група</i>		
Розробив	Більська А.Р.			24.02.23	<i>Вступ</i>			

доступність та ефективність навчання.

Аналіз попередніх досліджень. Дослідженню цифровізації освіти та розробки веобрієнтованих освітніх платформ присвячені праці вітчизняних та іноземних науковців: Л. В. Бондар, Т. А. Вакалюк, Т. М. Винарчук, Л. Е. Гризун, К. Ганое, Е. Дінк, І. І. Дончев, Дж. М. Керол, К. С. Куппусамі, Т. Л. Новицька, С. В. Новицький, М. Лі, А. Нурабаді, М. Б. Россон, А. М. Сущук, І. Ю Шахіна та ін.

Мета дослідження: розробка веборієнтованої навчальної платформи для гімназії з ціллю підвищення ефективності навчання та розширення можливостей освітнього процесу та розвитку цифровізації української освіти загалом.

Об'єкт дослідження: вебтехнології в освіті.

Предмет дослідження: розробка веборієнтованої навчальної платформи гімназії.

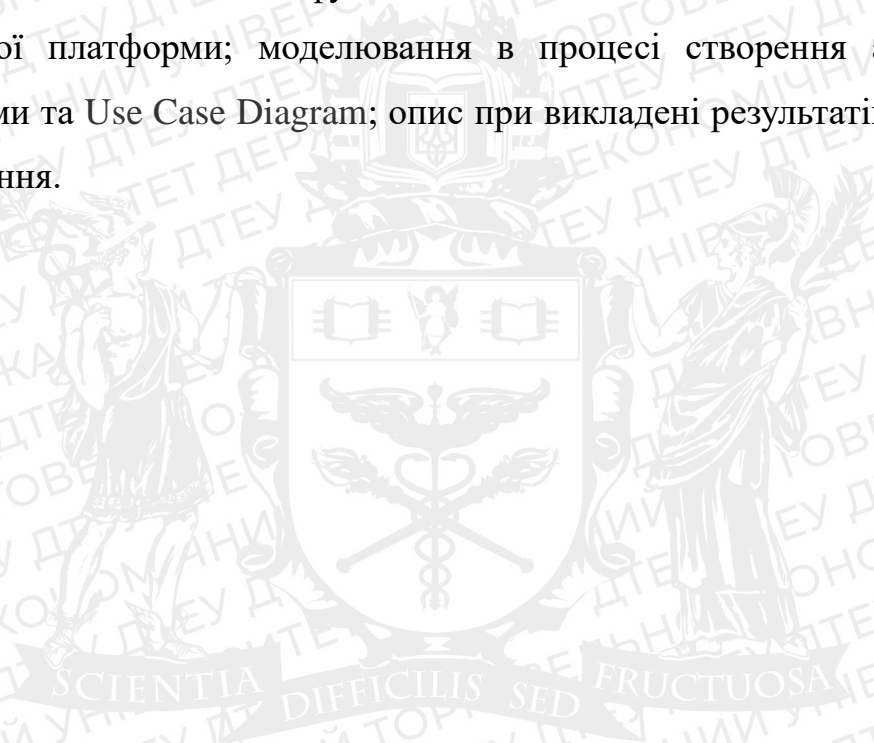
Завдання дослідження:

- Проаналізувати потреби та вимогу діджиталізованого освітнього процесу.
- Вивчити сучасні тенденції у веборієнтованих навчальних платформах.
- Розробити архітектуру веборієнтованої навчальної платформи гімназії.
- Оцінити результати проведеного дослідження та ефективність розробленої архітектури.

Наукова новизна дослідження. Наукова новизна дослідження полягає в розробці інноваційної веборієнтованої навчальної платформи, яка відповідає сучасним тенденціям у галузі освіти та забезпечує інтерактивність, доступність та ефективність навчання.

						Аркуш
						6
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-02.МР	

Методи дослідження: теоретичною та практичною основою дослідження виступили аналіз літературних джерел та наукових публікацій з питань веборієнтованих навчальних платформ та сучасних вебтехнологій. При виконанні роботи було використано метод порівняння наявних веборієнтованих навчальних платформ та технологій для їхньої розробки; аналізу і синтезу під час детального розгляду впровадження та інтеграції майбутніх технологій та функціональних особливостей веборієнтованої навчальної платформи; моделювання в процесі створення алгоритмічної блок-схеми та Use Case Diagram; опис при викладені результатів проведеного дослідження.



						Аркуш
						7
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-02.МР	

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1. Основні поняття веборієнтованої платформи

Традиційно програмне забезпечення було створене для певних платформ, таких як Windows, Linux або Mac OS. Однак сьогодні розробники активно працюють над створенням веб-додатків, які можуть працювати в Інтернеті незалежно від конкретної операційної системи користувача.

Вебплатформа є комплексом технологій, таких як HTML, CSS, JavaScript і інші відкриті стандарти, які використовуються для створення веб-сторінок. Вебплатформа постійно розвивається з метою поліпшення взаємодії з користувачем, забезпечення безпеки та конфіденційності. Зміни, внесені до вебплатформи, можуть вплинути на функціональність наявних веб-сторінок. [1].

Онлайн-платформа – це цифровий сервіс, який сприяє взаємодії між двома або більше різними, але взаємозалежними групами користувачів (організаціями чи окремими особами), які взаємодіють через сервіс через Інтернет. [2].

1.2. Порівняльний аналіз вебплатформ, комп'ютерних та мобільних застосунків

Веборієнтовані платформи мають ряд переваг та недоліків перед комп'ютерними та мобільними застосунками.

Десктопні додатки надають програмістам потужний контроль над проектуванням користувацького інтерфейсу. Завдяки графічним та візуальним інструментам програміст може створити точно такий інтерфейс, який

Зм.	Аркуш	№ докум.	Підпис	Дата	ДТЕУ 121 02з-02.МР			
Зав. каф.		Криворучко О.В.		10.04.23	Веборієнтована навчальна платформа гімназії	Стадія	Аркуш	Аркушів
Керівник		Котенко Н.О.		10.04.23		Р1	8	55
Гарант		Котенко Н.О.		10.04.23		Факультет інформаційних технологій 2мз курс, 2 група		
Розробив		Більська А.Р.		10.04.23				
					Аналіз предметної області			

потрібен, з меншим обсягом кодування. Однак, велика кількість сторонніх елементів у десктопних додатках може стати викликом, і вимагати витрат більшої кількості часових та грошових ресурсів на навчання та дослідження, щоб досягти більшої ефективності. З іншого боку, вебдодатки дозволяють програмістам використовувати користувацькі елементи, з якими більшість цільових користувачів вже знайомі. Десктопні додатки можуть бути складними і менш інтуїтивними для більшості користувачів.

Таблиця 1.1

ПЕРЕВАГИ ТА НЕДОЛІКИ ДЕСКТОПНИХ ТА ВЕБДОДАТКІВ

	Переваги	Недоліки
Десктопні застосунки	<ol style="list-style-type: none"> 1. Зазвичай, додатки можуть реалізувати ширший і більш просунутий функціонал. 2. Додатки дозволяють реалізувати функціонал нагадувань та повідомлень. 3. Користувачі мобільних девайсів надають перевагу використанню застосунків перед вебом. 	<ol style="list-style-type: none"> 1. Застосунки вимагають від пристроїв кінцевих користувачів певних технічних характеристик, наявності місця у сховищі телефона чи на диску користувача. 2. Низька мобільність. Необхідність у розробки для різних операційних систем та девайсів.
Вебзастосунки	<ol style="list-style-type: none"> 1. Програмісти можуть впроваджувати елементи, з якими більшість цільових користувачів вже знайомі, у функціонал та інтерфейс. 2. Велика портативність і доступність для користувачів та адаптивність. 3. Платформу легше 	<ol style="list-style-type: none"> 1. Велика залежність вигляду інтерфейсу і роботи деяких функцій від браузера. 2. Менш адаптивні для користування офлайн.

	<p>поширювати між користувачами.</p> <p>4. Вебплатформи легше оновлювати.</p> <p>5. Розробка вебзастосунків зазвичай дешевша.</p> <p>Вебплатформи дозволяють легше переміщуватись між вебресурсами, не залишаючи браузер.</p>	
--	---	--

Джерело: складено на основі [3], [4].

Таким чином, десктопні застосунки та веборієнтовані платформи мають ряд своїх недоліків та переваг, але виходячи з того, що перед нами стоїть завдання у розробці саме навчальної платформи і беручи до уваги нестабільну ситуацію в країні, зумовлену військовою агресією зі сторони російської федерації до України і високі вимоги до адаптивності та мобільності для школярів, батьків та вчителів, було обрано розробку саме веборієнтованої платформи.

1.3. Цифровізація в освіті

За визначенням енциклопедії з інформаційної науки та технологій, цифровізація означає впровадження цифрових технологій у різні аспекти повсякденного життя суспільства шляхом конвертації у цифровий формат всього, що є піддаємося цьому процесу. [9].

Згідно з доктриною «Україна 2030» цифрова трансформація включає в себе інтеграцію цифрових технологій у всі галузі бізнесу. Цей процес призводить до суттєвих змін у способах функціонування громадян, підприємств та організацій, а також у тому, як вони створюють цінність для себе, своїх співробітників, клієнтів та партнерів, здійснюють досягнення

					ДТЕУ 121 023-02.МР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		10

власних та спільних економічних та соціальних цілей швидше, ефективніше та з новою якістю. [10].

Сучасні інформаційні технології відкривають нові можливості для підвищення ефективності освітнього процесу та його вдосконалення. У сфері вищої освіти активно розробляється концепція дистанційної освіти, яка передбачає використання різноманітних технологій, включаючи змішане навчання. Згідно з цією концепцією, дистанційна освіта в Україні розглядається як рівноцінна форма навчання порівняно з традиційними формами, такими як очна, вечірня, заочна та екстернат. Вона реалізується в основному за допомогою технологій дистанційного навчання. [5].

Установлений Державним стандартом базової середньої освіти, затвердженим постановою Кабінету Міністрів України від 30 вересня 2020 року № 898 "Про деякі питання державних стандартів повної загальної середньої освіти", інформаційно-комунікаційна компетентність визначається як здатність впевнено, критично та відповідально використовувати цифрові технології для особистого розвитку і спілкування. Вона також передбачає безпечне використання інформаційно-комунікаційних засобів у навчанні та інших ситуаціях, дотримуючись принципів академічної доброчесності. [6].

Так, Європейський Парламент і Рада ЄС прийняли Рамкову програму оновлених ключових компетентностей для навчання протягом життя 22 травня 2018 року. У цій програмі цифрова компетентність визнана однією з восьми ключових компетентностей, необхідних для повноцінного життя та діяльності громадян ЄС. Угода про асоціацію між Україною, Європейським Союзом, Європейським співтовариством з атомної енергії та їх державами-членами, ратифікована Законом України від 16 вересня 2014 року № 1678-VII, передбачає сприяння розвитку співробітництва між сторонами в галузі освіти, навчання та молодіжної політики. [7].

						Аркуш
					ДТЕУ 121 023-02.МР	11
Зм.	Аркуш	№ докум	Підпис	Дата		

У Концепції "Нової української школи" велика увага приділяється інформатизації загальної середньої освіти. Можемо зробити висновок, що законодавчі ініціативи створюють основу для сприяння розвитку цифровізації освіти, онлайн-навчання та загального розширення діджитал-компетентностей викладачів та здобувачів освіти. Варто зазначити, що впровадження цифрових на інформаційних технологій ставить високі вимоги та вимагає глибоких досліджень у цій галузі зі сторони експертів. На цьому особливо наполягається у "Національному звіті про стан та перспективи розвитку освіти в Україні", а саме – «цифровізацію сфери освіти необхідно здійснювати на основі досягнень науково-технічного прогресу та психолого-педагогічної науки, реалізації в освітніх системах парадигм людиноцентризму і рівного доступу до якісної освіти, принципів відкритої освіти, широкого застосування в освіті сучасних цифрових технологій, поглиблення співпраці освітніх закладів, наукових установ і компаній ІТ-сфери у забезпеченні якісної освіти». [8].

Цифровізація освіти та впровадження інформаційних технологій у повсякденне навчання – наразі є однією з найважливіших та найстабільніших тенденцій розвитку світового освітнього процесу. Доведено, що застосування інформаційних технологій сприяє полегшенню та прискоренню процесу отримання знань, покращує якість сприйняття навчальних матеріалів і дозволяє краще розуміти та засвоювати їх. Цифрові інструменти, такі як комп'ютери, планшети, програмні засоби та онлайн-ресурси, надають учням можливість активно взаємодіяти з інформацією, розвивати критичне мислення та творчі здібності. Крім того, цифрові технології допомагають персоналізувати навчання, пристосовуючи його до індивідуальних потреб кожного учня. Інтерактивність, доступність різноманітних навчальних ресурсів та можливість негайного отримання зворотного зв'язку сприяють покращенню якості навчання і досягненню більш високих результатів серед

						Аркуш
						12
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-02.МР	

здобувачів освіти. Застосування цифрових технологій у навчанні є необхідною складовою для ефективної підготовки молодого покоління до сучасного інформаційного суспільства.

Популярність цифровізації серед педагогів пояснюється передусім її інтерактивністю, яка забезпечує швидку передачу інформації в обидва напрямки. В контексті навчального процесу, інтерактивність є ключовим елементом. У дистанційному навчанні, інтерактивність є необхідною під час роботи з різними програмами, електронними підручниками, базами даних та в процесі комунікації між учасниками навчання.

Цифровізація системи освіти спрямована на якісну перебудову. Її головна мета полягає в ефективному впровадженні нових інструментів та інформаційних ресурсів у навчальний процес, а також в "оцифруванні" освіти на основі таких базових технологій цифровізації, як мобільні комунікації та Інтернет. Це означає, що учні та вчителі мають мати доступ до сучасних мобільних пристроїв, які дозволяють швидко обмінюватися інформацією, отримувати доступ до різноманітних навчальних ресурсів та спілкуватися за допомогою онлайн-платформ. Така цифрова трансформація освіти допомагає забезпечити більш інтерактивне та зручне навчання, розширюючи можливості учнів і вчителів для активної участі у навчальному процесі. [11].

Віртуальні аудиторії та дистанційне навчання стали важливими інструментами електронного навчання на всіх рівнях шкільної освіти в США. Для якісного розвитку інноваційно-цифрової освіти важливо оновлювати освітні програми з фокусом на цифровізацію. Це передбачає використання сучасних освітніх технологічних інструментів, таких як інтерактивний інтерфейс, мультимедійний контент, онлайн-курси, онлайн-іспити, цифрові підручники, інтерактивні презентації, електронні конспекти та інше. Часто це все передбачає у собі функціонал навчальних платформ. [12].

						Аркуш
					ДТЕУ 121 023-02.МР	13
Зм.	Аркуш	№ докум	Підпис	Дата		

Таким чином, цифровізація освіти – це процес впровадження цифрових технологій у навчальну діяльність, який покращує якість навчання та сприяє розвитку інформаційно-комунікаційних компетенцій. Застосування інтерактивних цифрових інструментів у навчальному процесі сприяє підвищенню мотивації учнів та полегшує сприйняття навчального матеріалу. Цифрова трансформація освіти передбачає доступ до сучасних мобільних пристроїв та онлайн-ресурсів, що сприяє створенню більш інтерактивного та персоналізованого навчального середовища. Цифровізація освіти відіграє важливу роль у підготовці молодого покоління до сучасного інформаційного суспільства.

1.4. Висновок до розділу 1

Отже, в цьому розділі дипломної роботи було розглянуто важливі аспекти пов'язані з цифровізацією освіти та впровадженням веборієнтованих навчальних платформ. Дослідження показали, що вебплатформи мають деякі переваги перед звичайними десктопними застосунками, зокрема вони забезпечують більшу портативність, доступність для користувачів та адаптивність, що є визначними факторами для веборієнтованої навчальної платформи.

Також було проаналізовано державні програми з цифровізації в Україні, такі як доктрина "Україна 2030", Державний стандарт базової середньої освіти, Рамкова програма оновлених ключових компетентностей для навчання протягом життя та інші. Ці програми сприяють розвитку цифрових технологій в освіті та створюють сприятливе середовище для їх впровадження.

Особлива увага була приділена цифровізації в освіті та популяризації цифровізації серед педагогів. Сучасні освітні технологічні інструменти, такі як інтерактивний інтерфейс, мультимедійний контент, онлайн-курси, онлайн-іспити, цифрові підручники, інтерактивні презентації та електронні

						Аркуш
					ДТЕУ 121 023-02.МР	14
Зм.	Аркуш	№ докум	Підпис	Дата		

конспекти, стають необхідним елементом цифровізації освіти. Було виявлено, що цифрова трансформація освіти передбачає доступ до сучасних мобільних пристроїв та онлайн-ресурсів, що допомагає у створенні інтерактивного та персоналізованого освітнього процесу.

Було досліджено закордонний досвід впровадження цифровізації, а саме досвід США та використання сучасних веб-інтерактивних інструментів.

Отже, створення власної навчальної веборієнтованої платформи має великий потенціал для розвитку освіти в Україні. Вона дозволяє покращити якість навчання, залучити нових учасників освітнього процесу і забезпечити зручну та ефективну комунікацію між всіма сторонами. Це важливий крок у напрямку цифровізації освіти та підготовки учнів до майбутнього.



						Аркуш
					ДТЕУ 121 023-02.МР	15
Зм.	Аркуш	№ докум	Підпис	Дата		

РОЗДІЛ 2

ТЕХНОЛОГІЇ ТА ІНСТРУМЕНТИ ДЛЯ РОЗРОБКИ ВЕБОРІЄНТОВАНОЇ НАВЧАЛЬНОЇ ПЛАТФОРМИ ГІМНАЗІЇ

2.1. Аналіз та підбір Frontend технологій

2.1.1. Способи реалізації Frontend

Отже, як ми і проаналізували у попередньому розділі, навчальна платформа для гімназії має бути веборієнтованою, щоб забезпечити швидкий та легкий доступ до неї, швидкість оновлення більшу портативність та відносно невелику вартість розробки. Загалом веброзробку можна поділити на дві частини: Frontend та Backend або клієнтська та серверна частини.

Frontend розробка полягає у інтерфейсній розробці, а саме програмування та керування зовнішнім інтерфейсом вебсайту з метою забезпечення найкращого користувацького досвіду та інтерактивності вебплатформ та вебсайтів. Frontend розробка забезпечує функціональність різних ключових елементів вебсторінки та вебплатформи загалом, а також динамічності їхнього представлення у вигляді зручного та адаптивного інтерфейсу. В нашому випадку це може бути функціональність заповнення форм, проходження тестів, перегляду фото-галереї, електронної екскурсії гімназією. Окрім цього Front-End розробка передбачає реалізацію дизайну вебсайтів та вебплатформ, дотримання вимог до шрифтів, графіки та анімації. На додаток, Front-End розробка дозволяє налаштувати SEO та доступність, під'єднати Google analytics та інші сервіси для відслідковування користувацької активності на вебсайті. Таким чином, в нашому випадку, зі сторони Front-End будуть повністю або частко реалізовані такі

Зм.	Аркуш	№ докум.	Підпис	Дата	<i>ДТЕУ 121 023-02.МР</i>			
Зав. каф.		Криворучко О.В.		24.05.23	<i>Веборієнтована навчальна платформа гімназії</i>	Стадія	Аркуш	Аркушів
Керівник		Котенко Н.О.		24.05.23		P2	16	55
Гарант		Котенко Н.О.		24.05.23		<i>Факультет інформаційних технологій 2мз курс, 2 група</i>		
Розробив		Більська А.Р.		24.05.23				

функціонал як – галерея, особисті чати з викладачами та групові чати класів, вхід в кабінет та реєстрація, проведення тестувань, здачі робіт, онлайн календар подій, стрічка новин та інші можливості, якими вчителі, батьки, адміністрація гімназія та здобувачі освіти зможуть безпосередньо інтерактивно користуватися.

Загалом існує декілька основних видів створення вебзастосунків з точки зору Front-End.

а) Використання програмного забезпечення для створення вебсайтів.

Прикладами такого програмного забезпечення можуть бути платформа Wix, Wordpress та Squarespace. Такі платформи не вимагають знань з UX/UI дизайну, веброзробки, хостингу. Ці платформи мають готові шаблони та рішення для швидкого створення вебсайтів, а також функціонал для налаштування доступності та SEO. Платформа не вимагає жодних технічних знань, а вартість створення сайтів на цих платформах може бути безкоштовною або відносно недорогою. Однак цей спосіб створення вебсайтів має широкий список недоліків таких як обмежений функціонал та обмежене налаштування доступності та SEO, що не підійде для якісного просування та комфортного використання людям з обмеженими можливостями. Окрім цього функціонал даних платформ може не задовільнити вимоги дизайну і є доволі обмеженим для створення анімацій. Вебплатформи, створені за допомогою конструкторів зазвичай вимагають більше часу для завантаження, при цьому за статистикою 57% користувачів покидають вебсайт, якщо він не завантажився за 3 секунди, що каже про високі вимоги до швидкості роботи вебплатформ у користувачів. [13]. Ще одним вагомим недоліком є обмежена кількість вебсторінок, що є визначним фактором для веборієнтованої навчальної платформи. Незважаючи на всі ці недоліки, тим не менш, такі платформи все ще одними з найпопулярніших методів створення вебсайтів.

						Аркуш
						17
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-02.МР	

б) Використання стандартних технологій, інструментів та мов, таких як HTML, CSS, JavaScript. Це загально прийнятий метод, який показав себе добре, протягом багатьох років і досі є актуальним, проте втрачає свою популярність. Використання стандартного стеку Front-End технологій дозволяє реалізувати більшість функції, які передбачає проєкт будь-якої складності, починаючи від статичної лендінг сторінки, закінчуючи багатосторінковим продуктивним технологій. Тим не менш, не зважаючи на те, що ці технології дозволять розробити веборієнтовану навчальну платформу із повнішим функціоналом, ніж при використанні конструкторів сайтів, розробка займе значно більше часу. Крім цього цей спосіб вимагає не лише поглиблених знань і навичок програмування, а також знань із хостингу.

в) Розробка із використанням фреймворків. Цей спосіб створення вебзастосунків наразі є надзвичайно актуальним і популярним. Фреймворки дозволяють легко реалізувати всі можливості JavaScript та повністю або частково можливості HTML та CSS. Використовуючи цей метод розробки можна швидше розробити проєкт. Крім цього, це дозволяє легше структурувати проєкт, надає більше можливостей з точки зору доступних технологій, бібліотек, що пов'язані із даними фреймворками. Недоліками такої розробки можуть бути вимогливість до специфічних знань та вмінь в розробці, а також те, що дані фреймворки. Також, використання фреймворків сповільнює завантаження вебсайтів вперше, адже вимагає часу, щоб скомпілювати код у стандартні HTML JavaScript та CSS.

Таким чином, зважаючи на всі фактори у перелічених вище методах, нами було обрано розробку саме за використання фреймворків.

У наступних підрозділах розберемо стек технологій детальніше.

2.1.2. HTML та CSS

Незважаючи на те, що фреймворки можуть повністю або частково реалізовувати можливості HTML та/або CSS, веборієнтовані платформи все

						Аркуш
						18
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-02.МР	

ще вимагають наявності index.html файлу. Цей файл є необхідним для пошукових систем. HTML — це мова розмітки, яка визначає вміст, його структуру, в тому числі використовуючи семантичні теги. HTML складається з серії елементів, які дозволяють обгорнути частину змісту і надати йому певного вигляду та характеристик. [14]. Крім цього в тезі head у HTML-файлах розміщується така важлива інформація як метадані, title або заголовок сторінки для пошукових систем, стилі, скрипти, дані про те яка мова та версія HTML використовується, ключові слова, опис сторінки (або description) та інша важлива інформація, необхідна як для браузерів та пошукових систем, так і для кінцевих користувачів. Таким чином, проект обов'язково включатиме у себе index.html файл з усією визначною інформацією, яку вимагає тег head та SEO просування.

CSS. Каскадні таблиці стилів (CSS) — це простий механізм додавання стилю (наприклад, шрифтів, кольорів, інтервалів) до веб-документів. [15]. За допомогою CSS створюються анімації, сайт стає інтерактивнішим, дозволяє користувачу зрозуміти, що з елементом на сторінці можна взаємодіяти. Це можна використовувати, наприклад, за допомогою псевдокласу «:hover». Таким чином, при додаванні такого псевдокласу до кнопки, можна змінювати її колір. Користувач наводитиме курсором миші на кнопку і помічатиме, що відбулися певні зміни. Крім цього CSS також легко дозволяє зпозиціонувати елементи відносно потоку елементів або батьківських елементів. CSS також дозволяє з легкістю зберігати значення у змінних (такі як основні кольори, які використовуються у проектах, розміри та стани. А крім цього CSS дозволяє вираховувати певні дані за допомогою функції calc.

Щоб ефективніше працювати зі стилями ми використовуватимемо не просто CSS, а препроцесор SCSS, тобто Sassy Cascading Style Sheets, більш просунутий варіант CSS. SCSS дозволяє полегшити створення та використання змінних, вкладення класів, циклів, розширень для стилів та

						Аркуш
						19
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-02.МР	

багато іншого. Ці додаткові функції оптимізують CSS та роблять процес розробки набагато простішим і швидшим, ніж написання стандартного CSS. [16].

2.1.3. Порівняльний аналіз JavaScript та TypeScript

Зазвичай у веброзробці використовуються такі мови програмування як JavaScript та TypeScript.

JavaScript це не лише найвикористовуваніша мова програмування у веб-програмуванні, а й найбільш використовувана мова програмування у світі загалом. JavaScript це мова з динамічною типізацією. З 2015 року, коли було випущено ECMAScript, стандарт, затверджений міжнародною організацією ECMA, версії 6 (ES6), з'явилися нові функції, які значним чином розширили можливості JavaScript. ECMAScript постійно оновлюють версії, виправляють наявні проблеми, розширюють функціонал з попередніх версій.

TypeScript – це надмножина JavaScript, випущена корпорацією Майкрософт у 2012 році. Це означає, що TypeScript використовує всі існуючі функції ECMAScript, а також майбутні. TypeScript статично типізується та транспільується, тобто вихідний код перетворюється на JavaScript. TypeScript має ряд особливостей, таких як типізовані зміни, enum, union types. [17]. TypeScript, як і JavaScript підтримується більшістю сучасних фреймворків, таких як React, Vue.js, Angular та інші. Крім цього TypeScript дозволяє якісно та детально документувати код і робить його значно зрозумілішим для нових розробників, які приходять на вже існуючий проєкт.

Через свою динамічну та гнучку природу програми JavaScript часто мають репутацію програмного забезпечення низької якості. Згідно з дослідженнями TypeScript демонструє значно кращу якість коду та зрозумілість, ніж програми JavaScript. Однак код, написаний на TypeScript

						Аркуш
					ДТЕУ 121 023-02.МР	20
Зм.	Аркуш	№ докум	Підпис	Дата		

має вищу схильність до помилок на 60% і час їх, відповідно, в середньому більше часу для виправлення помилок. [18].

Таким чином TypeScript та JavaScript мови програмування, які добре показали себе у веброзробці і мають ряд своїх особливостей.

Таблиця 2.1

ПОРІВНЯННЯ JAVASCRIPT ТА TYPESCRIPT

JavaScript	TypeScript
<ul style="list-style-type: none"> - Мова програмування з динамічною типізацією; - Вимагає меншого часу для розробки; - Має обмежену типізацію; - Загалом проекти є менш зрозумілими; - Має ширшу документацію та екосистему; - Більше підходить для невеликих проєктів, які не вимагають постійної підтримки. 	<ul style="list-style-type: none"> - Мова програмування зі статичною типізацією; - Вимагає більшого часу для розробки; - Має розвинену типізацію; - Загалом дозволяє якісно документувати проекти; - Має меншу екосистему та вужчу документацію; - Більше підходить для складних проєктів, які вимагають постійної підтримки в майбутньому.

Таким чином можемо зробити висновок, що для реалізації навчальної веборієнтованої платформи гімназії більше підходить мова програмування TypeScript, адже, незважаючи на більшу складність розробки, вона дозволяє зробити проєкт зрозумілішим та уникнути неочевидних помилок, які можуть статися в майбутньому.

2.1.4. Порівняльний аналіз Frontend фреймворків. React

За даними опитування The 2022 State of JS найвикористовуванішими фреймворками у 2022 році були React, Angular та Vue.js. [19].

Розглянемо дані фреймворки детальніше.

						Аркуш
						21
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-02.МР	

а) React. React вже декілька років поспіль займає лідируючі позицію серед Frontend фреймворків. React першим реалізував унікальний синтаксис `.jsx`, що поєднує у собі логіку JavaScript та HTML верстку. Окрім цього при роботі з React також можливе використання TypeScript – в цьому випадку використовується синтаксис `.tsx`. React реалізує концепцію Virtual DOM, (Document Object Model), що дозволяє виконувати перерендеринг елементів на сторінці, що змінюються без перерендирування всієї сторінки загалом, що значно оптимізує роботу вебсайту. Крім цього React має у собі додаткові методи меморизації та загальної оптимізації, компонентний підхід, що додатково структурує код та можливість реалізації Single Page Application. React також має широкую екосистему, що дозволяє використовувати більшу кількість розширень та бібліотек, а також на базі React створено React Native – фреймворк, що використовується для створення мобільних застосунків для IOS та Android.

б) Angular – це інший популярний фреймворк для фронтенд-розробки, який також здобув велику популярність серед розробників. Angular надає розробникам потужні інструменти для створення веб-додатків. Він використовує TypeScript. Однією з ключових концепцій Angular є "система модулів", яка дозволяє організовувати код у логічні блоки. Ще одна з важливих особливостей Angular – "шаблонний синтаксис". За допомогою цього синтаксису розробники можуть описувати структуру користувацького інтерфейсу у вигляді шаблонів, які включають HTML та спеціальні директиви Angular. Це робить розробку користувацького інтерфейсу більш зрозумілою та простою. Angular також має вбудовану систему обробки подій, валідацію форм, маршрутизацію та інші функції. Angular надає можливість створення односторінкових додатків (Single Page Application) та розробку кросс-платформенних мобільних додатків за допомогою фреймворка Ionic.

						Аркуш
					ДТЕУ 121 023-02.МР	22
Зм.	Аркуш	№ докум	Підпис	Дата		

Значним недоліком Angular є його складність у порівнянні з іншими популярними фреймворками.

в) Vue.js – це ще один популярний фреймворк для Frontend-розробки, який швидко набирає популярність серед розробників. Подібно до Angular і React, Vue.js надає потужні інструменти для створення сучасних веб-додатків. Одним із ключових понять в Vue.js є "реактивність". Фреймворк відстежує залежності між даними та компонентами і автоматично оновлює відображення, якщо дані змінюються. У Vue.js також є свій унікальний шаблонний синтаксис, який включає в себе HTML-подібні шаблони з додатковими можливостями. Vue.js також пропонує модульну архітектуру, де додаток будується з окремих компонентів, які мають свою внутрішню логіку та можуть бути повторно використані. Це дозволяє структурувати код і підтримувати його в чистоті. Як і React та Angular – Vue.js надає можливість реалізації односторінкових додатків.

Таким чином, проаналізувавши три основні відмінності між React, Vue.js та Angular, ми обрали React, адже цей фреймворк є простим в освоєнні, широку екосистему та можливість швидко та легко адаптуватися для створення застосунків для IOS та Android в майбутньому.

Отже, ми визначились з основною мовою програмування – TypeScript та основним фреймворком – React – це будуть найосновніші технології та мови програмування зі сторони Frontend.

2.2. Аналіз та підбір Backend технологій. JavaScript, Node.js, Express.js

Наступною частиною веброзробки є Backend. Backend розробка, яку також називають серверною, відіграє важливу роль у процесах, що є невидимими для користувача, але необхідними для повноцінного використання вебсайтів. Backend відповідає за збереження та обмін даними між вебсайтом або базою даних програми та браузером. Крім цього Backend часто відповідає за безпеку даних, які отримуються та зберігаються та

						Аркуш
						23
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-02.МР	

безпеку від взлому чи атак на сервер. В нашому випадку, Backend може відповідати за такі задачі як: обробка запитів на реєстрацію та вхід користувачів через cookie-файли, збереження та публікацію актуальних даних, в тому числі і медіа-файлів, про гімназію, реалізацію виставлення, обробки оцінок, середнього балу, прорахування успішності учнів, збереження повідомлень та забезпечення їх доставки та відображення у веб-інтерфейсі, реалізація прорахування спеціальних «досягнень» в залежності від способів користування вебсайтом та інші елементи гейміфікації освіти. Таким чином Backend забезпечує обробку та збереження даних, взаємодію з базою даних, а також логіку та функціональність для основних елементів функціоналу вебсайтів та веборієнтованої навчальної платформи гімназії загалом.

За даними Next Technology Professionals, найпопулярнішими мовами програмування для вебзастосунків у Backend є JavaScript, Python та PHP. [20] Коротко порівнюємо ці мови програмування:

а) JavaScript. JavaScript є не лише відомою та зручною мовою програмування для Frontend, а й найпопулярнішою мовою для Backend розробки вебзастосунків. Один з популярних способів використання JavaScript на Backend - це використання платформи Node.js. Node.js - це середовище виконання JavaScript, яке побудоване на JavaScript-двигуні V8, що розробляється компанією Google. Воно дозволяє виконувати JavaScript-код на сервері, забезпечуючи високу продуктивність та можливість одночасно обробляти багато запитів. Основна перевага використання Node.js на Backend полягає у тому, використанні однієї мови програмування (JavaScript або TypeScript) як на Frontend, так і на Backend, що спрощує розробку і підтримку коду. Крім того, Node.js пропонує асинхронну модель програмування, яка дозволяє ефективно обробляти багатопотокові операції і

						Аркуш
					ДТЕУ 121 023-02.МР	24
Зм.	Аркуш	№ докум	Підпис	Дата		

забезпечувати високу продуктивність за рахунок неблокуючих операцій вводу/виводу (I/O).

б) Python. Python – це мова програмування, яку можна використовувати для різних видів проєктів, включаючи веб-додатки, наукові обчислення, штучний інтелект, автоматизацію завдань та багато іншого. Python має простий і зрозумілий синтаксис та підтримує об'єктно-орієнтоване програмування (ООП) та функціональне програмування, що дозволяє писати код у структурований та модульний спосіб. У сфері Backend розробки Python надає зручний набір інструментів для розробки вебдодатків, включаючи маршрутизацію, обробку запитів та відправку відповідей, а також забезпечує роботу з базами даних, аутентифікацію, авторизацію тощо. Однак незважаючи на широкий функціонал Python може бути повільнішим за JavaScript, використовувати більше пам'яті та мати більше синтаксичне навантаження.

в) PHP. PHP є мовою програмування, спеціально розробленою для веб-розробки. PHP має простий синтаксис та широкий набір вбудованих функцій і інструментів, спеціально призначених для веб-розробки. Це дозволяє легко обробляти запити, створювати і робити операції з базами даних, робити взаємодію з HTTP протоколом, розробляти сесії cookies тощо. PHP має широкий вибір розширень та фреймворків для роботи. Проте варто зазначити, що PHP це синхронна мова програмування, що має обмежену підтримку асинхронного програмування.

З урахуванням того, що основною мовою для Frontend розробки було обрано TypeScript, схожість цієї мови програмування з JavaScript, велику та широку екосистему та асинхронність, оберемо JavaScript як основну мову програмування для Backend.

Так само як і на Frontend, оберемо фреймворк для роботи з технологіями Backend.

						Аркуш
						25
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-02.МР	

Express.js є одним з найпопулярніших фреймворків для розробки веб-додатків на мові JavaScript. Вибір Express.js як фреймворку має свої обґрунтовані причини. Однією з ключових переваг Express.js є його простота та легкість використання. Фреймворк пропонує мінімалістичну архітектуру, що дозволяє розробникам швидко створювати веб-додатки без надмірних обмежень або складнощів. Express.js є дуже гнучким фреймворком, який дозволяє розробникам вибирати та налаштовувати компоненти додатку згідно з їхніми потребами. Отже, враховуючи всі ці фактори, оберемо Express.js у якості фреймворку для Backend нашого проєкту. [21]

2.3. Аналіз та підбір системи управління базами даних

Окрім Frontend так Backend, визначну роль у роботі веборієнтованої навчальної платформи буде відігравати система управління базою даних та безпосередньо база даних, де буде зберігатися інформація про користувачів платформи, пости, повідомлення у приватних чатах, зображення, оцінки та інша інформація – як результат прямого використання вебсайту.

Загалом існують дві моделі системи управління базами даних – реляційна (SQL) та нереляційна (NoSQL).

Реляційна модель даних базується на математичних теоріях, таких як теорія множин, теорія відношень та теорія логіки. Ця модель надає надійність, гнучкість і стабільність базі даних. Дані у реляційних базах даних зберігаються та організовані за допомогою формально описаних таблиць. Кожна таблиця має назву і складається з певного числа стовпців, які визначають типи даних, і рядків, які містять фактичні значення даних.

Нереляційні бази даних надають гнучку масштабованість, що означає, що вони можуть прозоро збільшуватися шляхом додавання нових вузлів і, зазвичай, розробляються з використанням доступного обладнання. [22].

Основна відмінність NoSQL баз даних від реляційних полягає в тому, що вони не використовують таблиці, стовпці та рядки для збереження даних.

						Аркуш
					ДТЕУ 121 023-02.МР	26
Зм.	Аркуш	№ докум	Підпис	Дата		

Замість цього, вони використовують різні структури даних, такі як ключ-значення (key-value), стовпці (column), документи (document) або графи (graph), для організації та збереження даних.

У випадку розробки веборієнтованої навчальної платформи гімназії ми оберемо реляційну базу даних, адже враховуючи загальні засади і принципи роботи SQL баз даних, вони мають у собі більш релевантні можливості, особливості та функціонал збереження даних для нашого проєкта, а саме:

- Структурованість даних – легка організація і взаємозв'язок структурованих даних, таких як інформація про школярів та викладачів, дані про академічну успішність у вигляді оцінок з певних дисциплін та ін.

- Складність запитів – SQL бази даних дозволяють зв'язування таблиці і виконувати агрегацію даних (наприклад, збереження досягнень користування платформою як елемент гейміфікації освіти, зв'язок між викладачем та учнями, учнем і оцінками та ін.).

- Реляційні бази даних мають вбудовану підтримку транзакцій, що дозволяє забезпечити цілісність даних. Це важливо для операцій, таких як здача робіт, прикріплення навчальних матеріалів і виставлення оцінок, де важливо гарантувати атомарність і консистентність даних.

Однією з найвикористовуваніших реляційних систем управління базами даних баз даних за 2022 рік є MySQL. MySQL є системою управління базами даних з відкритим кодом, що робить цю СУБД гнучкою та розширюваною. MySQL має простий у використанні синтаксис SQL, що дозволяє легко створювати, змінювати та запитувати дані в базі даних. А крім цього також має велику кількість документації, що полегшує роботу з базами даних. MySQL підтримує багато різних операцій системи Windows, Linux, macOS, що робить його універсальним і доступним для використання на різних платформах. Таким чином, нами було обрано саме MySQL у якості системи управління базами даних для веборієнтованої навчальної платформи

						Аркуш
					ДТЕУ 121 023-02.МР	27
Зм.	Аркуш	№ докум	Підпис	Дата		

гімназії. Розробимо алгоритмічну блок-схему, що демонструє операцію та загальний процес здачі домашньої роботи зі сторони учня вигляді прикріплення файлів (рис. 2.1.).

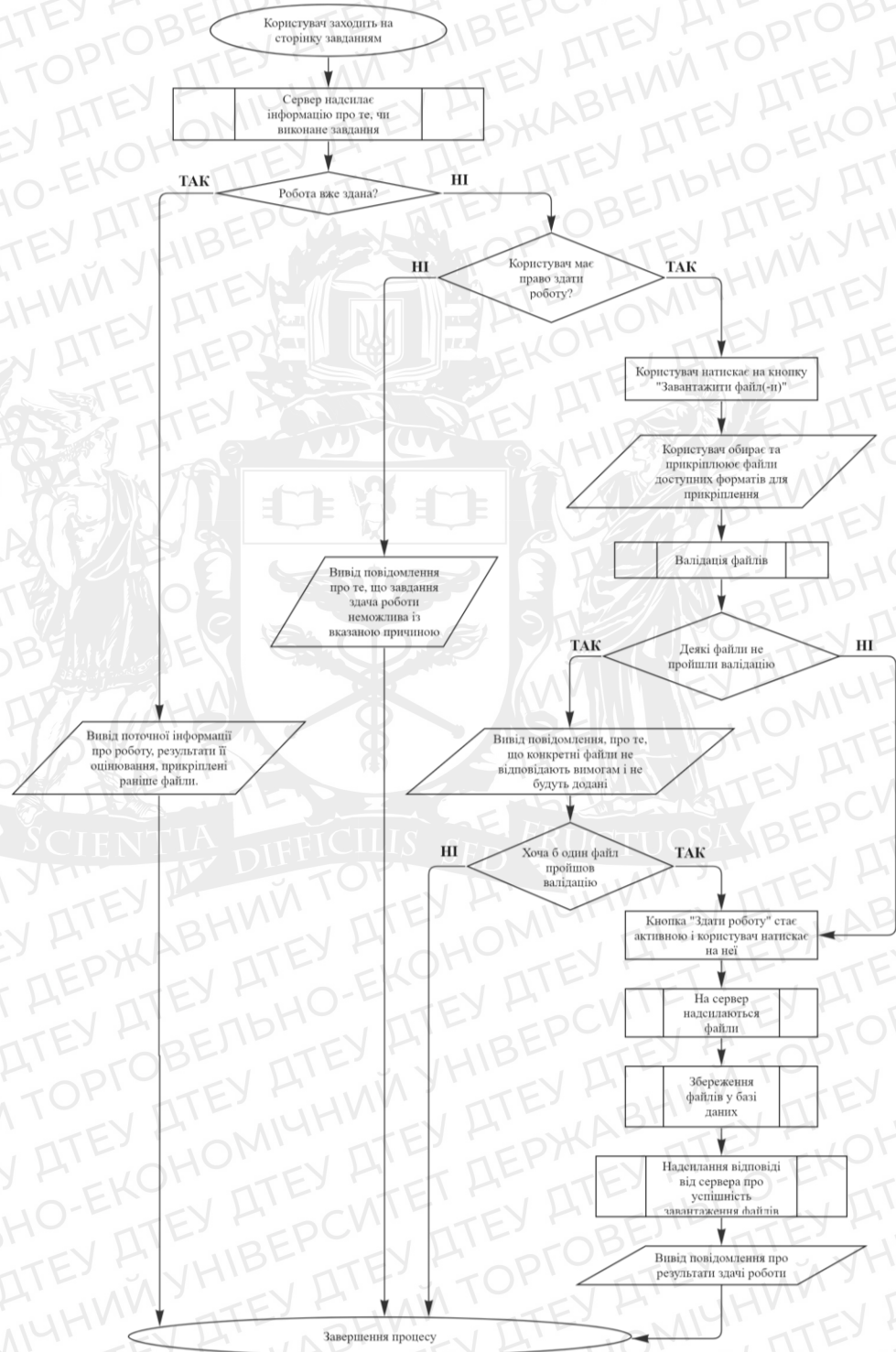


Рис. 2.1. Алгоритмічна блок-схема по процесу здачі домашньої роботи зі сторони учня

						Аркуш
						28
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-02.МР	

Таким чином, можемо побачити, що в процесі задачі домашньої роботи задіяні Frontend, Backend та База даних. Frontend, а саме TypeScript, HTML CSS та фреймворк React відповідальні за перші етапи відображення загального інтерфейсу, маршрутизація та опинення користувача на сторінці із певним завданням, а також блокування кнопки «здати роботу». На цьому ж етапі відбувається перевірка строків задачі роботи зі сторони Frontend. Frontend також відповідальний за прикріплення файлів та їхню валідацію, перевірку на відповідність доступним форматам, а також відповідність максимально-можливого розміру прикріплення файлів. Якщо файли валідні (або якщо хоча б один з прикріплених файлів валідний), зі сторони Frontend розблокується кнопка, що відповідає за задачу роботи, а саме відправки файлів на сервер та зміну статусу виконання роботи.

При цьому, при натисканні на розблоковану кнопку, буде надіслано запит на сервер про прикріплення файлів. Backend проєкту буде відповідальний за збереження файлів у базі даних та надсилання на клієнтську сторону повідомлення про успішне або неуспішне збереження файлів у базі даних. Таким чином, користувач завжди матиме доступ до файлів, які попередньо здав, а також вчителі будуть отримувати доступ до домашньої роботи своїх учнів.

На прикладі процесу задачі та перегляду виконання завдання домашньої роботи на навчальній веборієнтованій платформі гімназії, ми побачили, що більшість основних процесів будуть включати у себе як обрані нами основні технології Front-End, такі як HTML, CSS, TypeScript та React, так і технології Backend (JavaScript, Node.js, Express.js), а також інтеграцію СУБД MySQL.

2.4. Висновок до розділу 2

Отже, вибір основного стеку технологій для розробки веборієнтованої навчальної платформи гімназії є важливим етапом проєкту, оскільки від цього вибору залежить якість та функціональність системи. З метою

						Аркуш
						29
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-02.МР	

забезпечення найкращого веб досвіду для користувачів та оптимізації розробки, були обрані наступні технології.

У розділі клієнтської, або Front-End, розробки веборієнтованої навчальної платформи було вирішено використовувати TypeScript, React, HTML та CSS. TypeScript завдяки розвиненій типізації надає можливість виявляти та виправляти помилки на етапі компіляції, що покращує надійність коду та забезпечує легше управління проектом. React є бібліотекою, що дозволяє розробникам створювати ефективні та інтерактивні інтерфейси. Використання HTML та CSS дозволяє структурувати та стилізувати вміст, забезпечуючи зручне візуальне представлення.

У розділі Backend, або серверної, розробки вибрано JavaScript, Node.js та Express.js. JavaScript є популярною та масштабованою мовою програмування, що дозволяє реалізувати потрібну функціональність. Node.js надає зручну інфраструктуру для розробки серверних додатків, дозволяючи ефективно працювати з багатопотоковими запитами. Express.js, у свою чергу, забезпечує простоту та гнучкість у створенні API, що підтримує комунікацію між Frontend Backend.

Для зберігання даних обрано базу даних MySQL, яка є надійною та широко використовуваною системою керування базами даних. MySQL надає ефективне зберігання та обробку даних, що є необхідним для навчальної платформи гімназії.

Крім того, була створена та розглянута алгоритмічна блок-схема по задачі домашніх робіт, з виділеною роботою Frontend та Backend, а також була продемонстрована робота бази даних. Ця блок-схема допомагає систематизувати та упорядкувати процес виконання домашніх завдань, забезпечуючи чітке розуміння послідовності дій та взаємодії між різними компонентами системи.

						Аркуш
						30
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-02.МР	

Таким чином, обрані технології, такі як TypeScript, React, HTML, CSS, JavaScript, Node.js, express.js та СУБД MySQL, відповідають потребам розробки веборієнтованої навчальної платформи гімназії. Цей стек технологій забезпечує швидку та ефективну розробку, високу продуктивність та надійність системи. А на прикладі алгоритмічної блок-схеми ми побачили взаємозв'язок між усіма розділами та елементами розробки веборієнтованої навчальної платформи.



						Аркуш
					ДТЕУ 121 023-02.МР	31
Зм.	Аркуш	№ докум	Підпис	Дата		

РОЗДІЛ 3

РОЗРОБКА АРХІТЕКТУРИ ВЕБОРІЄНТОВАНОЇ НАВЧАЛЬНОЇ ПЛАТФОРМИ.

3.1. Інформаційна система веборієнтованої навчальної платформи

3.1.1. Предметна область та концепція

Згідно з інформацією, яку ми проаналізували у попередніх розділах, нам необхідно створити архітектуру веборієнтованої навчальної платформи.

Перед початком розробки архітектури проєкту, спроектуємо інформаційну систему.

Предметна область: веборієнтована навчальна платформа.

Назва Про: Розробка системи призначення, здачі та оцінювання домашніх робіт.

Назва ІС: Gymnasium.

LMS (Learning Management System) є основою веборієнтованої і забезпечує доступ вчителів та учнів до необхідних матеріалів та функцій. Вчитель, як адміністратор, має можливість створювати класи, призначати завдання та оцінювати їхні відповіді. Учні, які належать до конкретного класу, отримують доступ до завдань та можуть їх виконувати. Веборієнтована навчальна платформа має відповідати стандарту SCORM. Кожне завдання, або Sharable Content Object (SCO), може містити різні види контенту, такі як текст, відео, тести тощо. Ці SCO можуть бути легко призначені вчителем та виконані учнями, а після виконання вчителем проводиться оцінювання виконаних завдань. В режимі RTE вчителі та студенти мають змогу відслідковувати прогрес. Веборієнтована платформа має реалізовувати концепцію M-Learning.

Зм.	Аркуш	№ докум.	Підпис	Дата	ДТЕУ 121 023-2.МР			
Зав. каф.		Криворучко О.В.		06.09.23	Веборієнтована навчальна платформа гімназії	Стадія	Аркуш	Аркушів
Керівник		Котенко Н.О.		06.09.23		РЗ	32	55
Гарант		Котенко Н.О.		06.09.23		Факультет інформаційних технологій 2мз курс, 2 група		
Розробив		Більська А.Р.		06.09.23				
					Розробка архітектури веборієнтованої навчальної платформи.			

Веборієнтована навчальна платформа повинна надавати можливість:

- Створювати домашніх завдань вчителем для кожного окремого класу;
- Редагувати домашніх завдань вчителем;
- Додавати додаткові матеріали у вигляді файлів та/або посилань;
- Переглядати статистику прогресу перевірки завдань по кожному завданню для кожного класу вчителем;
- Переглядати статусу виконання домашнього завдання кожного окремого учня вчителем;
- Оцінювати домашні завдання вчителем;
- Переглядати список призначених домашніх завдань учнем.
- Прикріплювати виконані завдання у вигляді файлів та/або посилань.

Система має забезпечувати точність та достовірність обробки даних із зручним інтерфейсом для користувачів.

3.1.2. Нефункціональні вимоги до веборієнтованої навчальної платформи

- *Вимоги до застосування:*

1. Веборієнтована навчальна платформа має бути кросбраузерною;
2. ІС повинна бути адаптована і доступна з мобільних пристроїв, щоб користувачі могли мати доступ до неї з будь-якого місця та в будь-який час.

- *Вимоги до продуктивності:*

1. ІС повинна підтримувати мінімум сто одночасно працюючих з базою даних користувачів;
2. Інформаційна система має бути надійною та стабільною, щоб уникати втрат даних та непередбачуваних збоїв системи.

- *Вимоги до реалізації:*

1. Веборієнтована навчальна платформа повинна бути реалізована з використанням сучасних технологій та інструментів розробки.

						Аркуш
						33
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-02.МР	

2. Для забезпечення продуктивності та швидкодії системи має бути запити до бази даних мають бути оптимізовані.

- *Вимоги до інтерфейсу:*

1. Інтерфейс повинен бути інтуїтивно зрозумілим та легким у використанні.

2. *Вимоги до безпеки:*

1. Система повинна бути захищена від несанкціонованого доступу до даних користувачів.

3.1.3. Проєктна організація та опис бази даних

Для створення, призначення, перевірки та здачі домашніх робіт, а також формування класів, предметів та вчителів, було обрано рішення використовувати реляційну базу даних та використовувати MySQL у якості СУБД:

1. Таблиця «articles» (з англ. – статті). Таблиця матиме 4 поля – «id», «title», «description», «date», «img». Id є primary key.

id	title	description	date	img
1	Тиждень фізичної культури	Шановні учні та батьки! Цього року ми відзн...	2023-05-05	https://doc-14-6s-doc...
2	Випуск 2023	З надзвичайними емоціями та пишними надія...	2023-05-05	https://doc-0g-6s-doc...
NULL	NULL	NULL	NULL	NULL

Рис. 3.1. Таблиця «articles»

2. Таблиця «articles_tags» (з англ. – теги статей). Відобразатиме список усіх існуючих тегів для статей. Має дві колонки – id (primary key) та tag_name – назва тегу.

id	tag_name
1	Наша гордість
2	Спорт
3	Безпека

Рис. 3.2. Таблиця «articles_tags»

3. Таблиця «subjects» (з англ. – предмети). Таблиця матиме два поля – «subject_slug» та «display_text». Таблиця міститиме унікальний ідентифікатор у вигляді поля «subject_slug» та безпобесердньо повну назву

						Аркуш
						34
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-02.МР	

дисципліни українською мовою. У таблиці міститимуться всі предмети, які викладають в гімназії. Було обрано використовувати так званий slug у якості унікального строкового ідентифікатора, а не звичний числовий по причині обмеженості та унікальності предметів як таких. В даній таблиці subject_slug відіграє роль primary key.

subject_slug	display_text
ALG	Алгебра
AST	Астрономія
BIO	Біологія
DEU	Німецька мова

Рис. 3.3. Таблиця «subjects»

4. Таблиця «classes» (з англ. – класи). Таблиця має два поля – «class_id» та «class_name». class_id є головним ключем, тобто primary key, в той час як class_name репрезентує номер класу та його букву.

class_id	class_name
1000	7А
1001	7Б
1002	8А

Рис. 3.4. Таблиця «classes»

5. Таблиця «students» (з англ. – учні). Таблиця міститиме поля student_id, що репрезентує унікальний ідентифікатор учня, last_name, тобто прізвище учня, first_name ім'я, patronymic – по-батькові, phone_number – номер телефону у форматі 380XXXXXXXX, date_of_birth – дату народження, class_id – id класу, до якого відноситься учень, та picture, що містить і собі посилання на фотографію студента. Варто зазначити, що поля «picture» та «patronymic» можуть дорівнювати значенню null. Primary key у цьому випадку виступає student_id, в той час як class_id – це foreign key.

student_id	last_name	first_name	patronymic	phone_number	date_of_birth	class_id	picture
91	Корнієнко	Соломія	Олександрівна	380999876543	2009-02-10	1002	https://doc-04-6s...
92	Радченко	Стефан	Ігорович	380661245789	2009-06-29	1002	NULL
93	Бабій	Вероніка	Вікторівна	380506789123	2009-12-14	1002	NULL

Рис. 3.5. Таблиця «students»

6. Таблиця «teachers» (з англ. – вчителі). Таблиця містить поля teacher_id – що є унікальним ідентифікатором вчителя, а також – primary key; last_name, first_name, patronymic, що репрезентують прізвище, ім'я та по-батькові відповідно, phone_number – номер телефону у форматі 380XXXXXXXXXX, picture – є посиланням на зображення вчителя. Як і у випадку із таблицею «students», поля patronymic та picture можуть дорівнювати значенню null.


teacher_id	last_name	first_name	patronymic	phone_number	picture
9	Харлан	Антон	Васильович	380998769972	
10	Примаченко	Ангеліна	Едуардівна	380508723454	https://doc-0k-6s-docs.googleusercontent.com/docs/sec...
11	Міцкевич	Стефанія	Адамівна	380508724354	https://doc-0s-6s-docs.googleusercontent.com/docs/sec...
12	Патон	Валентин	Анатолійович	380963271776	https://doc-04-6s-docs.googleusercontent.com/docs/sec...

Рис. 3.6. Таблиця «teachers»

7. Таблиця «classes_subjects» (з англ. – класи-предмети). Ця таблиця репрезентує предмети, які викладаються у кожному класі, а також вчителів, які закріплені за цим предметом для кожного конкретного класу. Таким чином, таблиця має наступні поля: id – унікальний ідентифікатор для описаного вище співвідношення, class_id – є foreign key та використовується для позначення класу, у якому викладається дисципліна; subject_slug – також є foreign key та демонструє унікальний ідентифікатор предмета, який викладається; teacher_id – foreign key, що репрезентує вчителя, що викладає дисципліну у кожному окремому класі;

id	class_id	subject_slug	teacher_id
91	1002	ALG	3
92	1002	BIO	6
93	1002	DEU	18

Рис. 3.7. Таблиця «classes_subjects»

8. Таблиця «hometasks» (з англ. – домашні завдання). Таблиця має наступні поля – hw_id – primary key, що є унікальним ідентифікатором домашнього завдання; assign_to – foreign key, що посилається на id таблиці «classes_subjects»; title – назва домашнього завдання; description_text – опис, може дорівнювати null; is_required – демонструє чи є завдання обов'язковим

до виконання; *deadline* – демонструє термін здачі завдання; *max_mark* – показує яку максимальну оцінку студент може отримати за виконання завдання.

hw_id	assign_to	title	description_text	is_required	deadline	max_mark
1	91	Виконати впр...	Перед виконанням прочитайте ...	1	2023-09-11 23:59:59	11
2	98	Створити пре...	Тема: 1. Люблінська унія 1569р...	0	2023-09-14 23:59:59	12
3	115	Твір роздум на...	Написати від руки, сфотографу...	1	2023-09-14 23:59:59	11

Рис. 3.8. Таблиця «*hometasks*»

3.1.4. Дійові особи інформаційної системи

- *Вчитель*. Вчитель створює завдання, задає терміни відправки, прикріплює додаткові матеріали, вписує назву та опис завдання, а також оцінює виконані учнями завдання.
- *Учень*. Учень відправляє роботу на оцінювання, ознайомлюється з призначеними йому завданнями, приймає рішення стосовно виконання або невиконання необов'язкових завдань.
- *БД*. База даних зберігає інформацію про завдання, виконані завдання учнів та результати оцінювання.

3.1.5. Опис функціональних вимог

Здійснимо опис функціональних вимог веборієнтованої навчальної платформи гімназії на прикладі розділу «Домашні завдання», використовуючи Use Case Diagram мови UML. Для цього розподілимо сценарії між основними акторами. Варто зазначити, що при всіх перелічених нижче сценаріях буде задіяна база даних.

Сценарії для вчителя:

- Створення домашнього завдання для певного класу;
- Перегляд призначених завдань, загальної інформації та прогресу перевірки;
- Перегляд списку учнів, яким було призначень завдання, статус виконання, дату здачі та ін;

						Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-02.МР	
						37

- Виставлення оцінки за завдання.

Сценарії для учня:

- Перегляд списку завдань до виконання, разом з короткою інформацією про них;
- Здача виконаної роботи, а також перездача у разі незадовільної оцінки.

У вчителя та студента також буде спільний сценарій, а саме: перегляд розширеної інформації про завдання разом з прикріпленими додатковими матеріалами та роботою учня.

Зобразимо описані вище сценарії у Use Case діаграмі:

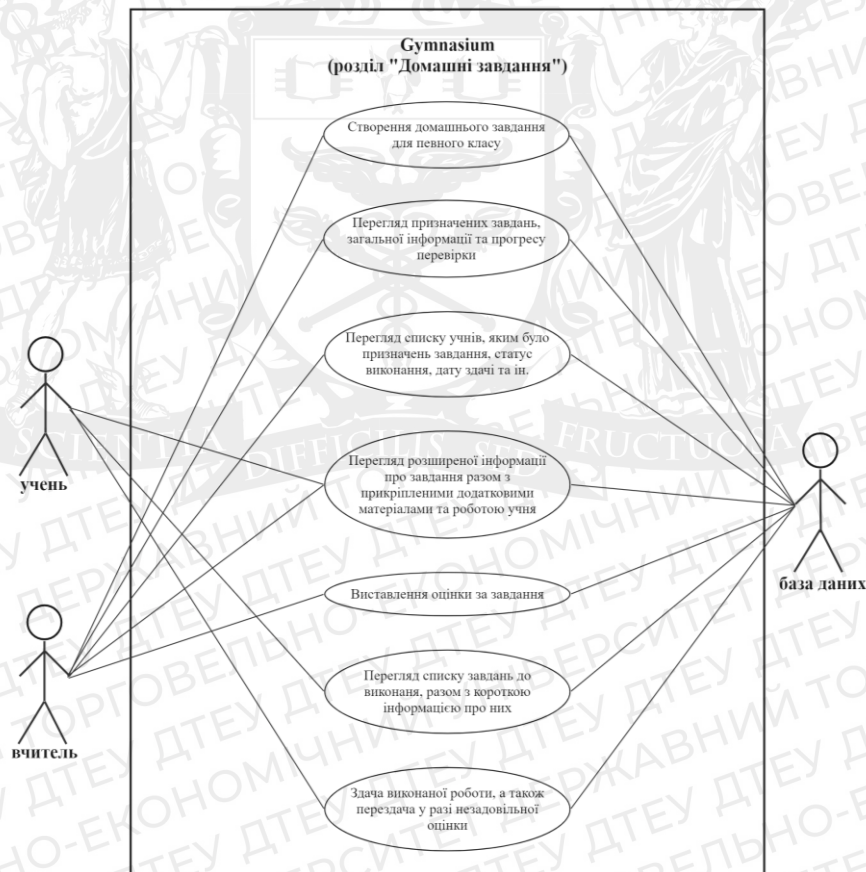


Рис. 3.9. Use Case Diagram веборієнтованої навчальної платформи, розділу «Домашні завдання»

						Аркуш
						38
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-02.МР	

3.2. Дизайн веборієнтованої навчальної платформи

Згідно з проаналізованою у попередньому розділі інформаційною системою, а також складеним технічним завданням, розробимо UX/UI дизайн веборієнтованої навчальної платформи. Для початку, розробимо wireframe гілки створення домашніх завдань, перегляду й оцінки для вчителя.

Шлях вчителя розпочинається із головної сторінки, на якій знаходиться хедер з кнопками, що ведуть до повідомлень, профілю користувача та бургер меню, в якому буде знаходитись розділ «Домашні завдання». Також на домашній сторінці знаходиться фото-слайдер та онлайн газета з основними шкільними новинами. При переході вчителем на сторінку «Домашні завдання», він зможе побачити список своїх призначень, тобто домашні завдання, які було призначено кожному окремому класі. У кожному елементі зі списку буде індикатор, що висвітлює відсоток перевірених домашніх завдань. На цій сторінці також буде знаходитись кнопка «створити», що вестиме на сторінку створення домашнього завдання.

При переході на сторінку створення домашнього завдання, перед вчителем відкриваються поля до заповнення. Серед них «клас» у вигляді випадачого списку, «назва», «опис», «дисципліна» у вигляді випадачого списку, «тип» - обов'язкове чи додаткове завдання, «термін» у вигляді дати, «максимальна оцінка» та кнопка для прикріплення додаткових матеріалів. Після заповнення усіх необхідних полів, користувач натискає на кнопку «створити» та переходить на сторінку із створеним домашнім завданням (попередній перегляд).

Після цього вчитель може повернутися на головну сторінку, де побачить нове завдання у списку призначень. Якщо вчитель натисне на кнопку «переглянути», що знаходить в елементі списку призначень, відкриється сторінка із загальним списком учнів, що належать класу, якому було призначене конкретне домашнє завдання. У цьому списку, користувач

						Аркуш
						39
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-02.МР	

побачить фотографію, ім'я і прізвище кожного учня, статус виконання ним домашньої роботи, дату здачі учнем домашньої роботи на перевірку, а також оцінку, яку поставив вчитель. Якщо ж оцінка не була виставлена, замість оцінки буде стояти прочерк. У разі, учень здав домашнє завдання, а вчитель його ще не переглянув, на фотографії учня з'явиться індикатор у вигляді кола, що повідомить вчителю про оновлення на сторінці домашнього завдання учня. Вчитель зможе сортувати учнів за ім'ям, оцінкою або даті здачі роботи. А також фільтрувати учнів в залежності від статусу виконання ними домашнього завдання, якщо натисне на відповідний статус. При кліку на ім'я учня, відкриється сторінка із домашнім завданням, на якій вчитель зможе оцінити домашнє завдання, якщо воно було попередньо здано. Вчитель також зможе змінити деякі параметри домашнього завдання, якщо відкриє детальну інформацію про нього.

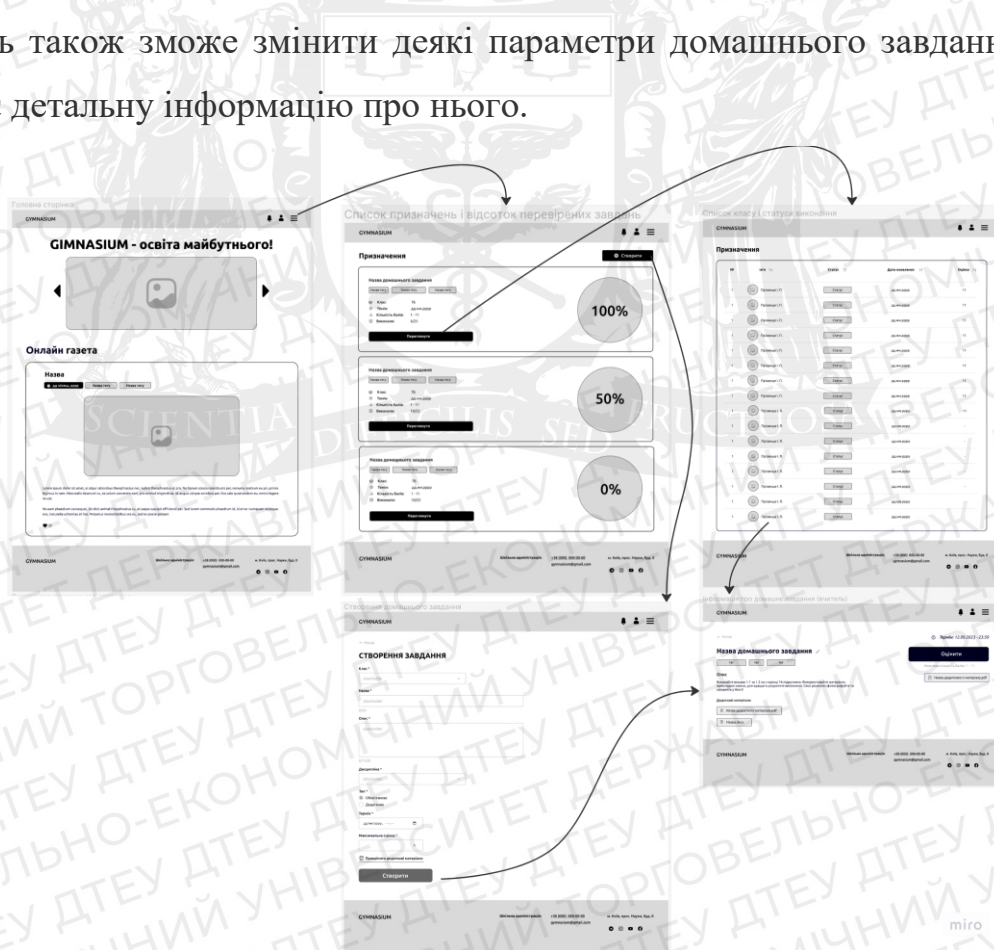


Рис. 3.10. Wireframe гілки створення та перевірки домашнього завдання вчителем

						Аркуш
					ДТЕУ 121 023-02.МР	40
Зм.	Аркуш	№ докум	Підпис	Дата		

Розробимо wireframe та проаналізуємо user flow, тобто шлях користувача зі сторони учня. Учень обиратиме розділ домашні завдання у бургер меню, тобто у боковому меню, яке відкриватиметься та закриватиметься при натисканні. Після цього користувач потраплятиме на сторінку зі списком доступних йому домашні завдань для виконання, а також завдань, що були виконані. Кожен елемент списку міститиме коротку інформацію про завдання, а саме, назву, теги до домашнього завдання, ПІБ вчителя, який призначив завдання, термін та максимальну кількість балів або оцінку в залежності від статусу виконання завдання. Крім цього кожен елемент списку домашнього завдання міститиме фотографію вчителя, що його призначив. При натисканні на кнопку «Переглянути», учень потрапить на сторінку із детальною інформацією про домашнє завдання, разом з описом та прикріпленими додатковими матеріалами вчителем. Учень також може прикріпити свої файли у цьому розділі, здати роботу, відмінити задачу, переглянути свою оцінку, якщо робота була попередньо оцінена та відмінити задачу. При оновленні інформації по кожному завданню, учень отримуватиме повідомлення у вигляді індикатора на фотографії вчителя на сторінці зі списком домашніх завдань.

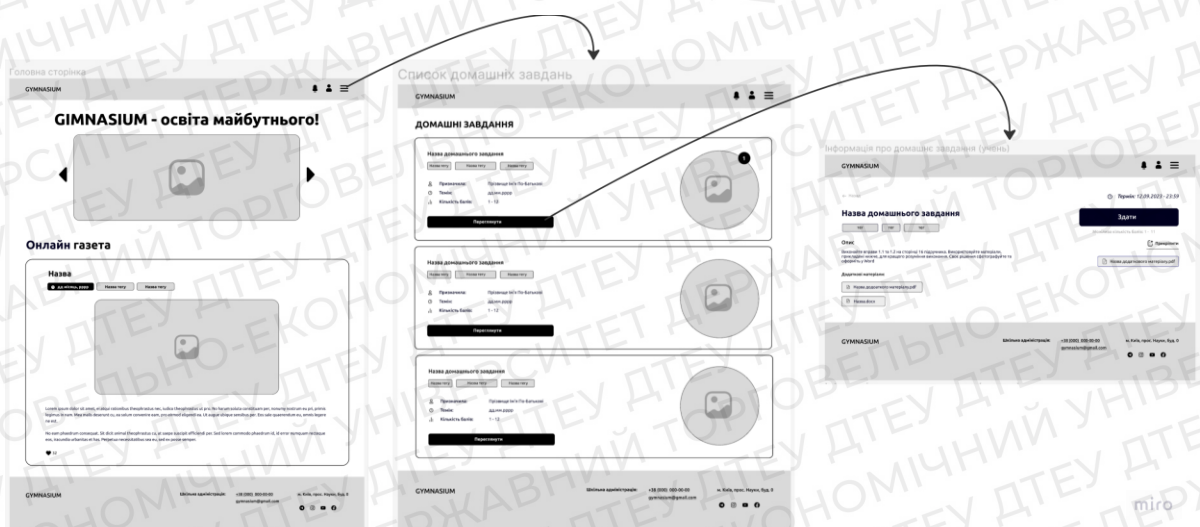


Рис. 3.11. Wireframe гілки створення та перевірки домашнього завдання учнем

Зм.	Аркуш	№ докум	Підпис	Дата
-----	-------	---------	--------	------

Таким чином, можемо детальніше розробити user interface з урахуванням корпоративних кольорів, шрифтів та детальнішого проопрацювання шляху користувача та користувачього досвіду.

Згідно з технічним завданням, основними корпоративними кольорами є #47C76B, #020077 та #FFFFFF. Ці кольори було використано в дизайні як основні. Крім цього в дизайні було обрано додаткові другорядні кольори. Згідно з вимог технічного завдання, було також зображено різноманітні можливі стани елементів, які користувачі можуть побачити в процесі використання. Таким чином, елемент списку призначень вчителя буде зеленим, якщо всі учні здали домашнє завдання і вчитель перевірів завдання кожного учня. Призначення буде червоним, якщо пройшло більше тижня з терміну здачі, але вчитель досі не перевірів всі здані домашні завдання. Та призначення буде білого кольору, якщо жодна з попередніх умов не виконана. Також, у демонстраційних цілях, в дизайн було додано можливі теги домашніх завдань. А саме: статус виконання, назва дисципліни, обов'язкове чи додаткове завдання.

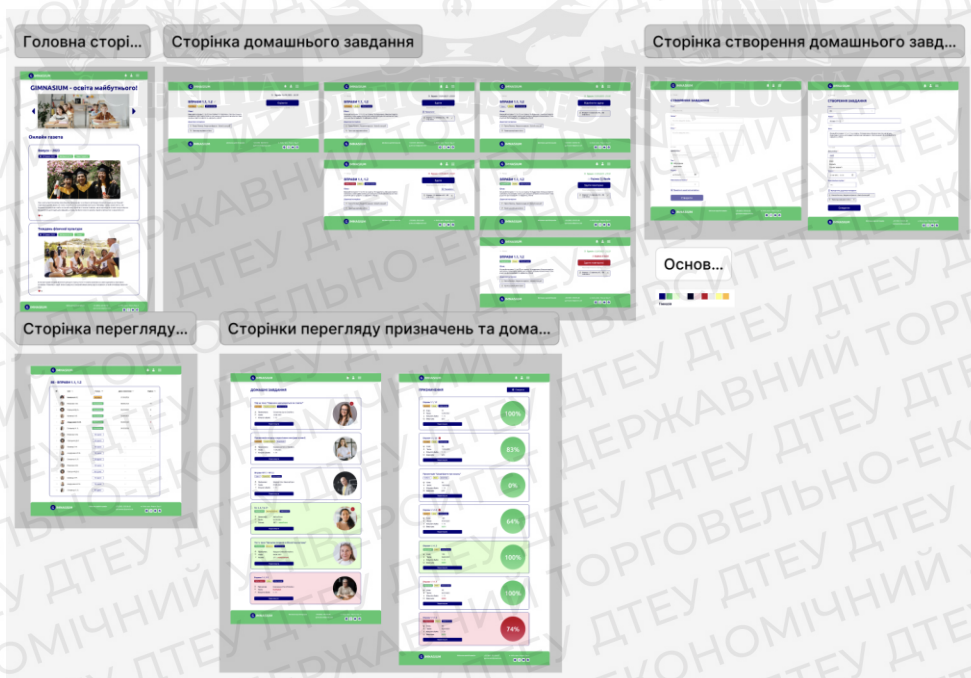


Рис. 3.12. Макет дизайну гілки створення та перевірки домашнього завдання учнем

						Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-02.МР	
					42	

3.3. Backend веборієнтованої навчальної платформи

Як було зазначено у попередніх розділах, для розробки Backend буде використовуватись `node.js`, мова програмування JavaScript та фреймворк `express.js`. Таким чином, для коректної роботи гілки задачі та перевірки домашніх робіт, а також домашньої сторінки, потрібно створити наступні методи:

1. Метод, що віддає статті для онлайн газети на головній сторінці;
2. Метод, що віддає призначення для вчителів;
3. Метод, що віддає список домашніх робіт для учнів;
4. Метод, що віддає список та статуси виконання учнів, що належать до певного класу, якому було призначено певне домашнє завдання – для вчителя;
5. Метод, який віддаватиме детальну інформацію по конкретному завданню;
6. Метод, який дозволить учням прикріпляти файли до домашнього завдання, зберігати їх в базі даних та тим самим помічати домашнє завдання як виконане;
7. Метод, який дозволить вчителям оцінити роботи та тим самим змінити її статус на «перевірено»;
8. Метод який дозволить створити нове призначення.

При проєктуванні backend веборієнтованої навчальної платформи використаємо архітектурний шаблон MVC, тобто Model-view-controller. Таким чином всі функції та операції на backend будуть розділені в залежності від свого призначення. Проєкт складається з 3 основних папок – `routes`, `services` та `controllers`. В кожній в цих папок знаходяться файли, які містять у собі функції для кожної з наявних сутностей, таким чином, файли, що містяться в папці `routes`, будуть називатися наступним чином – `teacher.route.js`, `student.route.js`, `homeworks.route.js`, `article.route.js`.

						Аркуш
					ДТЕУ 121 023-02.МР	43
Зм.	Аркуш	№ докум	Підпис	Дата		

В майбутньому кількість сутностей може зростати та збільшуватись, а операції для кожної з них будуть ставати все більш специфічними, саме тому було прийнято рішення розподілити наявні сутності таким чином для більшого розуміння коду та кращої структури проєкту.

Директива `controllers`, і відповідно логіка розроблена у файлах `teacher.controller.js`, `student.controller.js`, відповідає за обробку запитів, які надходять до веборієнтованої навчальної платформи від користувачів чи інших систем. Наприклад, у цій директиві будуть реалізовані методи по отриманню даних вчителів, учнів, домашніх завдань та ін. Також у контролері буде здійснена обробка помилок.

В директиві `services`, будуть знаходитись файли, що відповідають за обробку отриманих даних. Саме тут буде реалізована необхідна бізнес логіка. Сюди можуть входити різні обчислення, перетворення даних, валідація, логіка бізнес-правил та інші операції, які не є простою операцією читання або запису в базу даних.

Директива `routes` відповідає за те, як різні URL-запити мають бути оброблені. Маршрути визначають, які контролери повинні бути викликані для обробки різних запитів. В нашому випадку, маршрути будуть встановлені на рівні фреймворку `express.js`.

Щоб захиститися від SQL ін'єкцій, тобто типу атаки на безпеку, яка відбувається, коли зловмисники вводять зловмисний SQL-код в поля вводу або параметри запитів до бази даних, буде впроваджено такі заходи як: використання параметризованих запитів, екранування вхідних даних, а також валідація вхідних даних.

3.4. Frontend веборієнтованої навчальної платформи

Фронтенд ставить перед собою такі задачі, як реалізація UI, а саме верстка запропонованого дизайну по макету, підключення наявних методів

						Аркуш
					ДТЕУ 121 023-02.МР	44
Зм.	Аркуш	№ докум	Підпис	Дата		

до проєкту та впровадження якісного UX, шляхом оптимізації та налагодження всіх процесів.

Структура frontend проєкту матиме чотири основні директиви – pages, components, helpers та SCSSyles. Розглянемо за що відповідає кожна частина фронтенд веборієнтованої навчальної платформи.

Директива pages відповідає за функціональні компоненти, які використовуються у маршрутизації проєкту, тобто на які посилається React Router. Кожен функціональний компонент сторінок є батьківським елементом дочірніх елементів та об'єднує їх спільною логікою та структурою. Наприклад, контент сторінки Номерpage об'єднаний у компоненті Номерpage, що включає у себе такі компоненти як фото-слайдер, список статей шкільної газети та ін. При цьому варто зазначити, що routerView, тобто місце в проєкті в якому відображаються маршрути, буде розміщено в компоненті, у якому вже будуть підключені компоненти header, main та footer. Таким чином у кожному компоненті сторінок буде фактично заходитись контент блоку main та не буде потреби повторювати неунікальні компоненти header та footer.

Директива components буде відповідати за компоненти кожної сторінки. Варто зазначити, що як у випадку функціональних компонентів директиви «pages», так і у випадку функціональних компонентів директиви «components», компоненти будуть об'єднані в директиву, що відображає їхню назву. Наприклад, файл HomeworkList.tsx, буде знаходитись в папці HomeworkList разом з файлом HomeworkList.scss, що відповідає за стилі компоненту, та index.ts, що відповідає за зручний експорт.

Директива SCSSStyles, відповідає за підключення основних стилів проєкту. Стилї можна структурувати таким чином: основні кольори проєкту винесені у файл _colors.scss, шрифти – _typography.scss, нормалізація першопочаткового вигляду нативних елементів браузеру – _reset.scss.

						Аркуш
					ДТЕУ 121 023-02.МР	45
Зм.	Аркуш	№ докум	Підпис	Дата		

Міксини та плейсголдери scss зібрані у відповідні файли та також зберігаються у тій самій директиві. У файлі Mixins можна також прописати розміри основних екранів (mobile, tablet, desktop), що допоможе при налаштуванні адаптивної верстки в проєкті. Крім цього в директиві SCSSStyles знаходитимуться такі файли як: `_container.scss` для збереження основних відступів на сторінці та `_grid.scss` для розробки власної grid-сітки. Крім цього в даній директиві будуть зберігатися медіа файли, а саме фото і відео, які використовуються в UI та є статичними.

Директива helpers матиме у собі функції, які активно використовуються в проєкті, такі як `sleep`, `request` та ін.

Для state management, тобто процес керування даними і їх станами, використаємо бібліотеку `Redux`, що особливо добре поєднується із фреймворком `React`. `Redux` надає однорівневе централізований сховище для даних та набір правил для оновлення так званого стейту. У випадку проєкту феборієнтованої навчальної платформи, `Redux` допоможе уникнути "prop drilling", тобто передавання даних від батьківського елемента до непрямого дочірнього компонента, через багато компонентів. Це полегшить відстеження змін стану та дозволить уникнути повторного рендерингу деяких компонентів. В нашому випадку, `redux` може оперувати даними користувача.

Крім цього деякі дані будуть зберігатися у `local storage` та `session storage`, браузерному сховищі. Щоб забезпечити високий рівень безпеки веборієнтованої навчальної платформи та завадити зміненню збережуваних даних користувачем вручну, через `local storage` чи `session storage`, було прийнято рішення використовувати шифрування даних. Для цього використаємо бібліотеку `CryptoJS` та алгоритм шифрування `AES`.

						Аркуш
					ДТЕУ 121 023-02.МР	46
Зм.	Аркуш	№ докум	Підпис	Дата		

Для оптимізації роботи фронтенд веборієнтованої навчальної платформи ха зменшення навантаження на процесор, буде використано React.memo, useMemo та useCallback.

React.memo – це компонент вищого порядку, що допомагає уникнути повторного рендерингу компонента, якщо дані, які передаються у компонент від батьківського елемента (props) не змінилися.

useMemo – це хук, що дозволяє кешувати результати обчислень, щоб не повторювати їх при кожному рендері. Таким чином повторні обчислення будуть здійснені лише, якщо були змінені вихідні дані.

useCallback – це хук, який дозволяє кешувати колбек-функції, щоб уникнути створення нової функції при кожному рендері компонента. Це буде використано, при передачі колбеків дочірнім компонентам.

На Frontend буде активно задіяна валідація введених даних, особливо це буде стосуватися форми створення домашнього завдання. Для вчителя буде стояти обмеження у символах для назви та опису домашнього завдання, а також валідація прикріплених файлів (допустима кількість та формат). У разі, якщо введені дані не є валідними, користувач побачить повідомлення про це на екрані (вигляд повідомлення визначається завчасно підготовленим дизайном) за допомогою опціонального рендерингу.

3.5. Висновок до розділу 3

У веборієнтованій навчальній платформі "Gymnasium" визначено предметну область, створено глосарій термінів і описано концепцію системи. Система базується на Learning Management System (LMS) і відповідає стандарту SCORM, сприяючи створенню, призначенню, та оцінюванню домашніх завдань. Концепція включає Mobile Learning, забезпечуючи доступність на мобільних пристроях. Нефункціональні вимоги включають кросбраузерність, доступність на мобільних пристроях, продуктивність, стабільність, сучасні технології, інтуїтивний інтерфейс, безпеку та

						Аркуш
					ДТЕУ 121 023-02.МР	47
Зм.	Аркуш	№ докум	Підпис	Дата		

автентифікацію користувачів. Забезпечення точності та надійності обробки даних та зручного інтерфейсу для користувачів є ключовими завданнями для розробки даної навчальної платформи.

Для керування домашніми завданнями та інформацією про класи, предмети, учнів та вчителів було обрано використовувати реляційну базу даних і систему управління базами даних MySQL. Були розроблені таблиці для зберігання статей, тегів статей, предметів, класів, учнів, вчителів, відомостей про предмети в класах та домашні завдання. Кожна з цих таблиць має відповідну структуру та визначені ключі для забезпечення цілісності та зв'язків між даними.

У процесі розробки UX/UI дизайну для навчальної веборієнтованої платформи було створено wireframe для гілок створення та перегляду домашніх завдань як вчителем, так і учнем та домашньої сторінки. А також було створено фінальний макет дизайну з використанням корпоративних кольорів і шрифтів. Дизайн детально відображає шляхи користувачів та всі можливі сценарії використання, основні функції вчителя та учня зв'язані з створенням, виконанням та оцінкою домашніх завдань.

Розглянуто плани щодо розробки Backend для веборієнтованої навчальної платформи, використовуючи Node.js, JavaScript і Express.js. Визначено список методів, необхідних для реалізації функціоналу, таких як статті для головної сторінки, призначення для вчителів, список домашніх робіт і багато інших. Для структури проекту було обрано архітектурний шаблон MVC (Model-View-Controller), що допоможе розділити функціонал на компоненти.

Планується застосовувати заходи безпеки, такі як параметризовані запити, екранування вхідних даних і валідація, щоб запобігти SQL-ін'єкціям та іншим загрозам безпеці даних.

						Аркуш
					ДТЕУ 121 023-02.МР	48
Зм.	Аркуш	№ докум	Підпис	Дата		

Frontend проекту навчальної веборієнтованої платформи гімназії має за мету реалізацію користувацького інтерфейсу, використовуючи React, та забезпечення якісного користувацького досвіду через оптимізацію та впровадження Redux для керування станом додатку. Структура проекту включає компоненти сторінок та компоненти, які їх складають, оптимізовані для покращення продуктивності за допомогою React.memo, useMemo та useCallback. Також передбачено валідацію введених даних і шифрування для забезпечення безпеки користувачів.

Таким чином, розробили інформаційну схему веборієнтованої навчальної платформи, визначили головний план та напрямок дій та розробки, оптимізації продуктивності і безпеки Frontend, визначили основні методи заходів безпеки та структури Backend, створили макет дизайну (гілка створення та перевірка домашніх завдань) та розробили базу даних.

						Аркуш
					ДТЕУ 121 023-02.МР	49
Зм.	Аркуш	№ докум	Підпис	Дата		

ВИСНОВКИ ТА ПРОПОЗИЦІЇ

У результаті виконаної роботи, ми проаналізували теоретичні відомості про веборієнтовані навчальні платформи. Було виконаний порівняльний аналіз веборієнтованих платформ та десктоп застосунків. За результатом порівняльного аналізу, було надано перевагу реалізації веборієнтованій навчальній платформі, у зв'язку із високими вимогами до адаптивності та мобільності для школярів, батьків та вчителів.

Було з'ясовано останні тенденції у цифровізації освіти, а саме впровадження дистанційної освіти як альтернативному способу відвідування занять, використання віртуальних аудиторій, мережі інтернет та електронних пристроїв на заняттях, впровадження так званої гейміфікації у звичний навчальний процес. Було визначено необхідність глибоких досліджень в області цифровізації освіти задля якісної перебудови освітнього процесу.

Було проаналізовано та обрано інструменти для розробки Frontend. Після детального порівняльного аналізу методів та інструментів розробки Frontend, ми надали перевагу використанню фреймворку React, препроцесору SCSS, мови програмування TypeScript. Для інструментів розробки Backend було обрано мову програмування JavaScript, середу розробки Node.js та фреймворк Express.js. Для системи управління базами даних було надано перевагу використанню MySQL. Усі обрані технології було проаналізовано та порівняно з доступними альтернативами у розробці, визначено переваги та недоліки, з якими можна стикнутися при безпосередньому створенні веборієнтованої навчальної платформи гімназії.

Зм.	Аркуш	№ докум.	Підпис	Дата	ДТЕУ 121 02з-2.МР			
Зав. каф.		Криворучко О.В.		01.11.23	Веборієнтована навчальна платформа гімназії	Стадія	Аркуш	Аркушів
Керівник		Котенко Н.О.		01.11.23		ВП	50	55
Гарант		Котенко Н.О.		01.11.23		Факультет інформаційних технологій		
Розробив		Більська А.Р.		01.11.23		2мз курс, 2 група		
					Висновки та пропозиції			

Для наочної демонстрації взаємодії Backend, Frontend та бази даних, було спроектовано алгоритмічну схему, де відображався шлях користувача, із прикріпленням файлів та відправленням їх на перевірку вчителю. В результаті ми побачили явну взаємодію між всіма складовими веборієнтованої навчальної платформи.

Задля якісного створення та розробки архітектури веборієнтованої навчальної платформи, а саме гілки створення та перегляду домашніх завдань, було спроектовано інформаційну систему. Ми визначили предметну область, концепцію, нефункціональні та функціональні вимоги. У ході розробки інформаційної системи, було створено Use Case Diagram мови UML та описано базу даних.

Після детального проектування інформаційної системи, було розроблено дизайн веборієнтованої навчальної платформи гімназії. Задля цього було створено wireframes для шляху вчителя та учня, а після цього відображено повноцінний макет із дизайном. При створенні дизайну ми опиралися на вимоги, описані у технічному завданні, а саме – використання корпоративних кольорів та відображенні всіх можливих станів елементів, відображення валідації форм та ін.

При розробці Backend було визначено основні методи, необхідні для випуску проєкта, обрано архітектурний шаблон для реалізації, а саме архітектурний шаблон MVC, model-view-controller, структуру проєкту, заходи безпеки для уникнення атак у вигляді так званих SQL-ін'єкцій.

При розробці Frontend було розроблено структуру проєкту, було визначено необхідні елементи для створення та автоматизації стилів, а саме використання препроцесору SCSS та можливостей, які він надає: використання так званих mixin, placeholder та ін. На Frontend було обрано використовувати бібліотеку Redux для управління станами даних. Для оптимізації роботи веборієнтованої навчальної платформи гімназії, а також її

						Аркуш
					ДТЕУ 121 023-2.МР	51
Зм.	Аркуш	№ докум	Підпис	Дата		

продуктивності, було вирішено використовувати React.memo, useMemo та useCallback – хуки та компоненти, які є частиною фреймворку React. Для шифрування даних, було прийнято рішення використовувати бібліотеку Також передбачено валідацію введених даних і шифрування для забезпечення безпеки користувачів CryptoJS та алгоритм шифрування AES.

В майбутньому планується розширити функціонал веборієнтованої навчальної платформи гімназії, а саме, створити чіткий шлях авторизації, розділ сайту із віртуальною екскурсією навчальним закладом, систему проведення тестувань, систему нагородження за виконані завдання, так звані досягнення – частина процесу гейміфікації, реалізувати онлайн-щоденник та журнал з оцінками та домашніми завданнями, розмістити на веборієнтованій навчальній платформі розклад для кожного окремого користувача, що буде враховувати позакласні заняття для вчителів та учнів та ін.

В перспективі планується реалізувати застосунок за допомогою React Native, який вчителі, учні та батьки зможуть завантажити собі на телефон, використовуючи Google Play або AppStore.

Таким чином, у цьому дослідженні, розробили базу даних та дизайн, визначили необхідні технології та шляхи реалізації веборієнтованої навчальної платформи, проаналізували теоретичні відомості про цифровізацію освіти та визначили шлях розвитку проєкту.

						Аркуш
						52
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-2.МР	

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Development tips for Microsoft Edge // Режим доступу: <https://learn.microsoft.com/uk-ua/microsoft-edge/web-platform/web-platform> (дата звернення 05.06.2023).
2. Minati Sahoo, Karishma Rana Digital Education During COVID-19 in Odisha: Challenges and Prospects // Changing World Economic Order in the Post-Pandemic Period. – 2023.
3. Tehetena Masresha, Changing Desktop Application to Real Time Web Application // Metropolia University of Applied Sciences. – 2018. – 35 p.
4. Yoram Wurmser, The Majority of Americans’ Mobile Time Spent Takes Place in Apps. – URL : <https://www.insiderintelligence.com/content/the-majority-of-americans-mobile-time-spent-takes-place-in-apps> (дата звернення 05.06.2023).
5. Концепція розвитку дистанційної освіти в Україні. Постанова МОН України 20 грудня 2000 р. URL: <http://zakon.rada.gov.ua> (дата звернення 05.06.2023).
6. Державний стандарт базової середньої освіти. Постанова Кабінету Міністрів України 30 вересня 2020 р. URL: <https://zakon.rada.gov.ua/laws/show/898-2020-%D0%BF#n16> (дата звернення 05.06.2023).
7. Локшина О. І. Європейська довідкова рамка ключових компетентностей для навчання впродовж життя:

<i>ДТЕУ 121 023-2.МР</i>					
Зм.	Аркуш	№ докум.	Підпис	Дата	
Зав. каф.		Криворучко О.В.		01.11.23	
Керівник		Котенко Н.О.		01.11.23	
Гарант		Котенко Н.О.		01.11.23	
Розробив		Більська А.Р.		01.11.23	
<i>Веборієнтована навчальна платформа гімназії</i>					
<i>Список використаних джерел</i>					
			Стадія	Аркуш	Аркушів
			СВД	53	55
			Факультет інформаційних технологій 2мз курс, 2 група		

оновлене бачення 2018 року // Український педагогічний журнал. – 2019.

– №3. – С. 21-30.

8. Лапаєнко С. В. Теоретико-методологічне забезпечення цифрової трансформації освіти і педагогіки // Інноваційна педагогіка. – 2023. – №55. С. 9-13.
9. Encyclopedia of Information Science and Technology, Fourth Edition (10 Volumes). IGI Global. – 2017. – 8104 p.
10. Український інститут майбутнього. URL: <https://strategy.uifuture.org/kraina-z-rozvinutoyu-cifrovoyu-ekonomikoju.html> (дата звернення 05.06.2023).
11. Гуралюк А.Г. Цифровізація як умова розвитку системи освіти // Вісник Національного університету «Чернігівський колегіум» імені Т.Г. Шевченка. – 2021. – №13. С. 3-8.
12. Кравченко С. М. Цифровізація як тенденція розвитку шкільної освіти у США // IV Міжнародна конференція "Педагогічна компаративістика і міжнародна освіта 2020". – 2020. – С. 56-57.
13. Shaun Anderson, How fast should a website load in 2023. URL: <https://www.hobo-web.co.uk/your-website-design-should-load-in-4-seconds/> (дата звернення 05.06.2023).
14. HTML basics // MDN Web Docs. URL: https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/HTML_basics (дата звернення 05.06.2023).
15. Cascading Style Sheets home page // W3C. URL: <https://www.w3.org/Style/CSS/Overview.en.html> (дата звернення 05.06.2023).
16. Sass Documentation. URL: <https://sass-lang.com/documentation> (дата звернення 05.06.2023).

						Аркуш
						54
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-2.МР	

17. Manuel Merkel Do TypeScript Applications Show Better Software Quality than JavaScript Applications? A Repository Mining Study on GitHub / Dr. Justus Bogner, Prof. Dr. Stefan Wagner // Thesis, University of Stuttgart, Stuttgart, Germany. – 2021. – 75 p.
18. Justus Bogner To type or not to type?: a systematic comparison of the software quality of JavaScript and typescript applications on GitHub / Manuel Merkel // International Conference on Mining Software Repositories. – 2022. – №19. P. 658-669.
19. Front-end Frameworks / Sacha Greif, Eric Burel // The 2022 State of JS survey. URL: <https://2022.stateofjs.com/en-US/libraries/front-end-frameworks> (дата звернення 05.06.2023).
20. Top 7 Backend Programming Languages in 2022, NEXT TECHNOLOGY PROFESSIONALS. URL: <https://nexttechnology.io/top-7-backend-programming-languages-in-2022/> (дата звернення 05.06.2023).
21. Express.js Documentation. URL: <https://expressjs.com/> (дата звернення 05.06.2023).
22. Ковтун Б. В. Порівняльна характеристика реляційних та NoSQL баз даних / Б. В. Ковтун, А. М. Манич, О. В. Романюк // Вінницький національний технічний університет. – 2023. – 3 с.

						Аркуш
						55
Зм.	Аркуш	№ докум	Підпис	Дата	<i>ДТЕУ 121 023-2.МР</i>	

ТЕХНІЧНЕ ЗАВДАННЯ

Таким чином можемо скласти технічне завдання.

Назва: "Gymnasium".

Версія: 1.0.

Мета: у першій версії розробити дизайн та архітектуру веборієнтованої навчальної платформи для гімназії, яка має у собі функціонал розповсюдження інформації про гімназії, а також призначення, здачу та оцінювання домашніх завдань.

Цільова аудиторія:

- Здобувачі освіти гімназії, діти віком від 6 до 18 років;
- Вчителі гімназії;
- Адміністрація гімназії;
- Батьки та/або опікуни здобувачів освіти.

Проблематика: наявний шкільний сайт гімназії застарілий, не має розширеного, окрім як інформаційного функціоналу, не оптимізований для мобільних пристроїв. На сайті не оптимізована структура та відсутні будь-які налаштування доступності для користувачів з обмеженими можливостями. Вебсайт має застарілий дизайн, що негативно впливає на імідж гімназії.

Опис вебплатформи та вирішення наявних проблем: розробка нової вебплатформи із функціоналом створення домашніх завдань, їхньої здачі та оцінки, а також інформаційним функціоналом. Вебплатформа має мати сучасний та зручний дизайн, функціонал аутентифікації, мобільну та

					<i>ДТЕУ 121 023-2.МР</i>			
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
Зав. каф.		Криворучко О.В.		06.09.23	<i>Веборієнтована навчальна платформа гімназії</i>	<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
Керівник		Котенко Н.О.		06.09.23		<i>ТЗ</i>	<i>56</i>	<i>55</i>
Гарант		Котенко Н.О.		06.09.23		<i>Факультет інформаційних технологій 2мз курс, 2 група</i>		
Розробив		Більська А.Р.		06.09.23				
					<i>Технічне завдання</i>			

десктопну версії.

Детальний опис та шлях реалізації:

а) Розробка бази даних. База даних має містити інформацію про учнів, класи, до яких вони відносяться, предмети доступні до вивчення, інформацію про вчителів, зібрані домашні завдання та медіа файли, які до них відносяться. Крім цього у базі даних мають міститися новини гімназії.

б) Розробка дизайну. Дизайн має містити основні корпоративні кольори, а саме: #47C76B, #020077 та #FFFFFF. Дизайн має бути розроблений для головної домашньої сторінки, на якій відобразатимуться новини гімназії, сторінка входу в кабінет, сторінка із переліком домашніх завдань зі сторони учня, сторінка із детальною інформацією про домашнє завдання, сторінка із демонстрацією призначень зі сторони вчителя, сторінка прогресу виконання домашнього завдання кожним конкретним класом із переліком учнів цього класу. Сторінка перегляду призначеного завдання зі сторони вчителя, сторінка створення нового завдання зі сторони вчителя. Дизайн має чітко демонструвати всі можливі стани кожної зі сторінок та елементів, наприклад, наявність протермінованих завдань, відсутність домашніх завдань, успішно виконані завдання і т.д.

в) Розробка методів backend. Методи мають реалізувати функціонал аутентифікації, віддавати інформацію про новини сайту, наявні домашні завдання учня, детальну інформацію про завдання як зі сторони вчителя, так і зі сторони учня, список призначень для вчителя, прогрес виконання завдань учням, перегляд, видалення та оцінку виконаного завдання вчителем, а також створення нових завдань за допомогою CRUD операцій.

г) Розробка frontend – а саме верстка та підключення методів, розробка роутингу між сторінками, валідація форм.

На кожному етапі має бути впроваджено тестування та оцінка якості виконання завдань проєкту згідно ви

						Аркуш
						57
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-2.МР	

ДОДАТКИ

Додаток А

Аналіз та порівняння веборієнтованих навчальних платформ

Цифрове освітнє середовище базується на низці принципів, які гарантують ефективність та зручність для користувачів. Основні принципи включають доступність та відкритість, системність, інтерактивність, поліфункціональність та орієнтованість на користувача.

Принцип доступності та відкритості означає, що цифрове освітнє середовище повинне бути доступним для всіх користувачів без обмежень, незалежно від їхніх можливостей та особливостей. Воно повинне підтримувати стандарти доступності, щоб забезпечити рівний доступ до навчальних ресурсів та інструментів.

Принцип системності передбачає, що цифрове освітнє середовище має бути організоване в логічну та послідовну систему, де всі компоненти взаємодіють між собою. Це допомагає забезпечити послідовність та структурованість навчального процесу, а також легкість навігації та орієнтації для користувачів.

Принцип інтерактивності вказує на необхідність взаємодії між користувачем та цифровим освітнім середовищем. Воно повинне забезпечувати можливості для активної участі, взаємодії та співпраці між учнями, викладачами та іншими учасниками навчального процесу.

Принцип поліфункціональності передбачає, що цифрове освітнє середовище повинне підтримувати широкий спектр функцій та можливостей. Воно повинне включати різноманітні інструменти, ресурси та додаткові функції, які дозволяють учням та викладачам виконувати різні завдання, надавати зворотний зв'язок та сприяти навчанню.

Принцип орієнтованості на користувача означає, що цифрове освітнє середовище повинне бути розроблене з урахуванням потреб, інтересів та

можливостей користувачів. Воно повинне бути зручним, інтуїтивно зрозумілим та адаптованим до різних стилів навчання та індивідуальних вимог користувачів.

Враховуючи ці принципи, цифрове освітнє середовище може стати ефективним інструментом для навчання та розвитку, сприяючи залученню учнів та поліпшенню їхніх навчальних результатів.

Розглянемо найпопулярніші навчальні веборієнтовані платформи, що найбільше відповідають вищезгаданім принципам. Умовно можемо поділити веборієнтовані навчальні платформи на дві категорії: платформи для організації навчання, включаючи дистанційне навчання, і платформи для отримання практичних та теоретичних знань.

Перша категорія платформ зазвичай має у собі функціонал демонстрації академічної успішності здобувачів освіти (електронні щоденники та журнали), календаря із розкладом занять, стрічку новин для публікації повідомлень про події, які відбудуться у закладі освіти а також чати для спілкування між студентами та викладачами. Подібні платформи можуть включати у себе також і функції тестування, прикріплення навчальних матеріалів та здачі й оцінення вчителями робіт школярів. Ці платформи також можуть інтегруватися з сервісами відео-спілкування або мати власні відео-конференції. Такі платформи можуть бути як і загальними (Moodle, Microsoft Teams, Google Classroom) так і розробленими для конкретних закладів освіти.

Друга категорія – навчальні онлайн-платформи для безпосереднього отримання теоретичних знань та навичок. Ця категорія платформ зазвичай використовується як додаткове джерело отримання знань та навичок, містить у собі специфічні курси, які в свою чергу включають в себе велику кількість відео-, аудіо- та текстових матеріалів. У цих платформах практично не ведеться реальна комунікація між викладачами та студентами, оцінка знань відбувається за допомогою тестування. До того ж користування навчальними матеріалами може бути платним. До таких платформ можна віднести

Всеукраїнську Школу Онлайн, Prometheus, Coursera та ін. Використання навчальних платформ може бути корисним додатковим інструментом для навчання, допомагаючи студентам отримати доступ до різноманітних ресурсів та покращити свої навчальні результати.

З огляду на більшу відповідність принципам цифрового освітнього середовища, ми спиратимемось саме на першу категорію веборієнтованих навчальних платформ, однак майбутня навчальна платформа також буде частково включати у себе елементи та функціонал другої категорії.

Отже розглянемо та порівняємо найпопулярніші навчальні платформи, які активно використовуються у закладах освіти України.

а) **Google Classroom.** Google Classroom - це безкоштовна веб-платформа, розроблена для зручного розподілу завдань та ефективної комунікації між студентами та викладачами. Вона тісно пов'язана з широким спектром сервісів Google, таких як Google Sheets, Google Docs, Google Meet, Google Drive, Google Calendar та інші. Ця платформа надає всі необхідні інструменти для успішної організації навчання, легкого доступу до навчальних матеріалів та виконання завдань. Крім цього, Google Classroom також пропонує можливість використовувати його офлайн у обмеженому режимі, що є важливим аспектом, особливо в умовах воєнного положення. Платформа дозволяє створювати окремі класи, додавати навчальні матеріали, спілкуватися з учнями та виставляти оцінки. Крім того, вона має інтегрований календар, що сприяє більш організованому плануванню навчального процесу. Однак варто відзначити, що на платформі відсутній журнал оцінок, що може становити певне обмеження у веденні детальної оцінювальної документації. Загалом, Google Classroom є потужним інструментом для організації навчального процесу, забезпечуючи зручну взаємодію між студентами та викладачами, але варто враховувати його обмеження в частині журналу оцінок. [2].

б) **Moodle.** Moodle є однією з найвідоміших платформ для організації онлайн-навчання, яка широко використовується багатьма вищими та середніми навчальними закладами. Вона відмінно допомагає у комплексній організації дистанційного навчання. Moodle надає можливість завантажувати матеріали, проводити тестування і, загалом, має ширший функціонал порівняно з Google Classroom. Однак варто відзначити, що для використання Moodle необхідний власний хостинг і домен. Хоча сама програма є безкоштовною, обслуговування хостингу зазвичай потребує платних послуг. Крім того, навчальний заклад повинен забезпечити навчання персоналу у використанні платформи, що може вимагати додаткових часових та фінансових витрат. Загалом, Moodle є потужним інструментом для організації навчального процесу в онлайн-середовищі, проте варто враховувати технічні вимоги та необхідність навчання персоналу для ефективного використання цієї платформи. [3].

в) **Microsoft Teams.** Microsoft Teams, подібно до Google Classroom, має широку екосистему, включаючи власний сервіс відеодзвінків, календар, можливість додавати матеріали і багато іншого. Крім того, Teams має функціонал особистих чатів і журнал з оцінками. Загалом, цей сервіс надає більші можливості для роботи, ніж Google Classroom, але менші, ніж Moodle, і він менш автономний. Однак слід врахувати, що для використання Teams потрібен обліковий запис Microsoft, і в умовах не завжди стабільного Інтернету він може бути менш надійним. [4].

Таким чином ці навчальні платформи мають список своїх переваг та недоліків, але їх всіх об'єднує фактор недостатньої автономії, залежність від інших програм та складність впровадження власного функціонала. Наприклад, жоден з даних сервісів не передбачає впровадження гейміфікації в освіті.

Гейміфікації в освітньому процесі використовує ігрові елементи або принципи гри для досягнення певних цілей. Основна перевага гейміфікації полягає в її мотиваційних можливостях. Використання ігор в освіті вже довело свою ефективність, оскільки граючись, учні засвоюють більше інформації і краще її запам'ятовують. При дистанційному навчанні гейміфікація може бути використана для ознайомлення користувачів з системою управління навчанням або для залучення учнів до активної участі в навчальному процесі. Один з прикладів сервісів, які допомагають впровадити гейміфікацію, - це Kahoot - онлайн-платформа, яка дозволяє створювати інтерактивні навчальні ігри, такі як вікторини, обговорення і опитування. [1].

При використанні гейміфікації на вебплатформі можна також впровадити систему досягнень, яка стимулює користувачів до активності та прогресу в навчанні. Ця система базується на принципі нагородження учнів за досягнення певних цілей або виконання завдань.

Наприклад, вебплатформа може надавати віртуальні значки, медалі, рівні або бейджі за успішне виконання завдань, активну участь у форумах, виконання додаткових вправ чи проходження віртуальних курсів. Кожне досягнення може мати свою унікальну назву та опис, що робить процес нагородження більш цікавим та мотивуючим для учнів.

Подібна система досягнень може відобразитись у профілі користувача на платформі, де вони можуть переглядати свої успіхи, зібрані нагороди та прогрес у навчанні. Крім того, платформа може створити рейтингову систему, де учні можуть порівнювати свої досягнення з іншими учасниками та спонукати один одного до активності та зростання.

Така система досягнень на вебплатформі не лише створює конкурентну та мотивуючу атмосферу, але й допомагає вчителям відстежувати прогрес учнів, визначати їхні сильні та слабкі сторони та індивідуалізувати навчальний процес для кожного учня. На жаль, жодна з перелічених вище загальних веобрієнтованих навчальних платформ не має у собі подібного функціоналу, який розкривався би повною мірою.

Окрім цього власна навчальна веб-платформа відіграє важливу роль в маркетингу і брендингу навчального закладу. Фактор наявності онлайн-платформи для навчання та комунікації сприятиме наступному:

а) Власна навчальна платформа дозволить не лише оновити та цифровізувати навчальний процес, а й залучити нових здобувачів освіти, вчителів та педагогів. Адже за допомогою власної навчальної платформи можна поширити на широку аудиторію цінності закладу освіти, провести віртуальну екскурсію, повідомити про останні заклади які проводились у школі та познайомити із загальним освітнім процесом та особливостями навчання.

б) Навчальна платформа дозволяє школярам, вчителям та батькам завжди залишатися на зв'язку та швидко поширювати інформацію яка стосується як освітнього процесу, так і безпосередні навчальні матеріали. Це особливо важливо в умовах воєнного часу, коли ситуація в країні нестабільна. До того ж спільний канал зв'язку дозволяє легше поширювати інформацію не лише серед учнів певного класу чи певної дисципліни, а й на весь персонал освітнього закладу та учнів.

в) Власна навчальна платформа дозволяє об'єднати здобувачів освіти вчителів та навчальну спільноту. Власна навчальна платформа також є інструментом брендингу та організації власного корпоративного стилю.

г) Наявність сучасної та актуальної вебплатформи школи дає батькам впевненість, що вони віддали своїх дітей у школу, яка забезпечить високий рівень освіти та підготовки до майбутнього. Інвестування часу та грошових ресурсів у створення інформативної, функціональної шкільної вебплатформи показує батькам існуючих та потенційних учнів, що заклад освіти відповідає сучасним тенденціям та цифровізації. [5] .

Такі маркетингові заходи повністю або частково неможливі при використанні загальнодоступних навчальних платформ.

Список використаних джерел додатку А

1. Худавердієва В. Тенденції цифрової трансформації освіти в сучасних умовах // Педагогічні науки та освіта. – 2022. – №XL-XLI. С. 102-109.
2. About Classroom. URL: <https://support.google.com/edu/classroom/answer/6020279?hl=en> (дата звернення 05.06.2023).
3. Moodle Documentation. URL: https://docs.moodle.org/402/en/Main_page (дата звернення 05.06.2023).
4. What is Microsoft Teams. URL: <https://support.microsoft.com/en-us/topic/what-is-microsoft-teams-3de4d369-0167-8def-b93b-0eb5286d7a29> (дата звернення 05.06.2023).
5. Niche Ignite, Reasons Why Your School Needs a Website. URL: <https://nicheignite.com.au/school-websites/5-reasons-why-your-school-needs-a-website/> (дата звернення 05.06.2023).