

Вінницький національний технічний університет

Факультет інформаційних технологій та комп'ютерної інженерії

Кафедра програмного забезпечення

Бакалаврська дипломна робота

на тему: «Розробка веб-системи моніторингу та підтримки навчально-виховного процесу у закладах освіти І ступеня»

Виконав: студент __ 4 __ курсу

групи 3ПІ-186

спеціальності

121 – Інженерія програмного забезпечення

(шифр і назва напрямку підготовки, спеціальності)

Лаба Д.С.

(прізвище та ініціали)

Керівник: к.т.н., доц. каф. ПЗ Войтко В.В.

(прізвище та ініціали)

Рецензент: к.т.н., доц. каф. КН Арсенюк І.Р.

(прізвище та ініціали)

Допущено до захисту

Зав. кафедри _____

« ____ » _____ 2022 р.

Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра програмного забезпечення
Рівень вищої освіти перший бакалаврський
Галузь знань 12 – Інформаційні технології
Спеціальність 121 – Інженерія програмного забезпечення
Освітньо-професійна програма – Інженерія програмного забезпечення

УЗГОДЖЕНО
Завідувач Малинківського НВК
Турчанівська Н. П.
25 березня 2022 р.

ЗАТВЕРДЖУЮ
Завідувач кафедри ПЗ
Романюк О.Н.
25 березня 2022 р.

З А В Д А Н Н Я НА БАКАЛАВРСЬКУ ДИПЛОМНУ РОБОТУ СТУДЕНТУ

Лабі Діані Сергіївні

1. Тема роботи – «Розробка веб-системи моніторингу та підтримки навчально-виховного процесу у закладах освіти І ступеня»

Керівник роботи: Войтко Вікторія Володимирівна, к.т.н., доц. кафедри ПЗ, затверджені наказом вищого навчального закладу від 24 березня 2022 р. № 66.

2. Строк подання студентом роботи 13 червня 2022 р.

3. Вихідні дані до роботи: середовище розробки Visual Studio Code, мова розробки JavaScript, операційна система – Windows 7, система управління базою даних MongoDB, базові алгоритми.

4. Зміст розрахунково-пояснювальної записки: вступ; аналіз та постановка задачі аналіз даних; розробка інтерфейсу веб-системи; розробка методу моніторингу та підтримки навчально-виховного комплексу; розробка моделі веб-системи; розробка алгоритмів веб-системи; розробка програмного забезпечення веб-системи; тестування веб-системи; висновки; перелік посилань; додатки.

5. Перелік графічного матеріалу: мета, об'єкт та предмет дослідження, задачі дослідження, наукова новизна, практична цінність одержаних результатів, порівняльний аналіз аналогів, графічний інтерфейс веб-системи; метод

моніторингу та підтримки навчально-виховного комплексу; модель роботи веб-системи; блок-схеми алгоритмів роботи веб-системи; тестування веб-системи.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада Консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1-4	Войтко В. В., к.т.н., доцент кафедри ПЗ		

7. Дата видачі завдання 25 березня 2022 р.

КАЛЕНДАРНИЙ ПЛАН

з/п	Назва етапів бакалаврської дипломної роботи	Строк виконання етапів роботи	Примітка
	Аналіз веб-систем для моніторингу навчально-виховного комплексу та вибір чітких цілей для поставленої задачі	26.03.2022 – 02.04.2022	Вик.
	Аналіз і вибір мови програмування та середовища розробки	05.03.2022 – 11.04.2022	Вик.
	Розробка алгоритмів веб-системи	15.04.2022 – 23.04.2022	Вик.
	Розробка модуля електронного журналу	06.05.2022 – 20.05.2022	Вик.
	Тестування веб-системи	21.05.2022 – 01.06. 2022	Вик.
	Оформлення матеріалів до захисту БДР	02.06.2020 – 10.06.2022	Вик.

Студент

(підпис)

Лаба Д.С.
(прізвище та ініціали)

Керівник бакалаврської дипломної роботи

(підпис)

Войтко В.В.
(прізвище та ініціали)

АНОТАЦІЯ

Бакалаврська дипломна робота складається з 57 сторінок формату А4, на яких є 40 рисунків, 3 таблиця, список використаних джерел містить 27 найменувань

У бакалаврській дипломній роботі проведено детальний аналіз аналогів, які існують, визначено їх переваги і недоліки. Сформульовано мету досліджень – підвищення ефективності роботи закладу освіти І ступеня шляхом розробки спеціалізованої веб-системи з засобами моніторингу та підтримки навчально-виховного процесу в молодшій школі, що забезпечить ефективну взаємодію суб'єктів освітнього процесу.

Запропоновано метод моніторингу та підтримки навчально-виховного процесу, цей метод передбачає, що у вчителя є можливість мати доступ до електронного журналу, в якому акумульована інформація про учнів, їх місце проживання, контакти батьків, а також до сторінок журналу з оцінками по предметах, в яких кожен з педагогів має виставляти оцінки учням. Щоб зрозуміти, як працює веб-система для моніторингу та підтримки навчально-виховного комплексу, було розроблено модель роботи системи.

Розроблено алгоритми, структуру інтерфейсу веб-ресурсу, метод моніторингу та підтримки навчально-виховного процесу в закладах освіти.

Отримані в бакалаврській дипломній роботі результати можна використати для моніторингу та підтримки навчально-виховного комплексу І ст.

Програмний продукт було створено з використанням мови програмування JavaScript. В якості середовища розробки програмного забезпечення було обрано Microsoft Visual Studio Code.

Ключові слова: веб-система, веб-сайт, електронний журнал.

ANNOTATION

The bachelor's thesis consists of 57 pages A4, which have 40 figures, 3 tables, a list of sources used contains 27 items

In the bachelor's thesis a detailed analysis of existing analogues, identified their advantages and disadvantages. The purpose of the research is to increase the efficiency of the first-degree educational institution by developing a specialized web system with tools for monitoring and supporting the educational process in primary school, which will ensure effective interaction of the subjects of the educational process.

A method of monitoring and supporting the educational process is proposed, this method assumes that the teacher has access to an electronic journal, which accumulates information about students, their place of residence, parents' contacts, as well as pages of the journal with grades in subjects which each of the teachers must grade students. To understand how the web system for monitoring and supporting the educational complex works, a model of the system operation was developed.

Algorithms, structure of the web resource interface, method of monitoring and supporting the educational process in educational institutions have been developed.

The results obtained in the bachelor's thesis can be used to monitor and support the educational complex of the first century.

The software product was created using the JavaScript programming language. Microsoft Visual Studio Code was chosen as the software development environment.

Keywords: web system, website, electronic journal.

ЗМІСТ

ВСТУП	8
1 ОБҐРУНТУВАННЯ ВИБОРУ МЕТОДУ РОЗРОБКИ ТА ПОСТАНОВКА ЗАДАЧ ДОСЛІДЖЕННЯ	11
1.1 Аналіз стану веб-систем для моніторингу та підтримки навчально- виховного процесу у закладах освіти.....	11
1.2 Порівняльний аналіз аналогів.....	13
1.3 Аналіз методів розв’язання задачі.....	18
1.4 Постановка задач для веб-системи моніторингу та підтримки навчально- виховного процесу	21
1.5 Висновки	21
2 РОЗРОБКА МЕТОДУ, МОДЕЛІ ТА АЛГОРИТМІВ РОБОТИ ВЕБ- СИСТЕМИ	23
2.1 Аналіз інформаційного забезпечення веб-системи	23
2.2 Розробка структури інтерфейсу веб-системи.....	24
2.3 Розробка методу моніторингу та підтримки навчально-виховного процесу	27
2.4 Розробка моделі роботи веб-системи.....	28
2.5 Розробка алгоритмів роботи веб-системи	29
2.6 Висновки	33
3 ПРОГРАМНА РЕАЛІЗАЦІЯ ВЕБ-СИСТЕМИ ДЛЯ МОНІТОРИНГУ ТА ПІДТРИМКИ НАВЧАЛЬНО-ВИХОВНОГО ПРОЦЕСУ	34
3.1 Варіантний аналіз і обґрунтування вибору засобів для реалізації програмного засобу.....	34
3.2 Розробка бази даних.....	39
3.3 Розробка модуля електронного журналу.....	41
3.4 Розробка клієнтської частини	42
3.5 Розробка головного файлу сервера	43

3.6 Висновки	44
4 ТЕСТУВАННЯ ВЕБ-СИСТЕМИ	45
4.1 Тестування веб-системи	45
4.2 Розробка інструкції користувача	56
4.3 Висновки	57
ВИСНОВКИ.....	58
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	59
ДОДАТКИ.....	62
Додаток А – Технічне завдання	63
Додаток Б – Акт впровадження	67
Додаток В – Протокол перевірки на плагіат	69
Додаток Г – Лістинг програми.....	71
Додаток Д – Графічна частина.....	90

ВСТУП

Обґрунтування вибору теми дослідження. У сучасному світі розвиток технологій призводить до розширення системи Інтернет-комунікацій. Нові підходи до навчального процесу вимагають сучасних рішень. У наш час кожен освітній заклад повинен мати власну веб-систему підтримки навчально-виховного процесу, яка надає широкі функціональні можливості і підтверджує, що заклад освіти йде в ногу з часом.

Веб-система та веб-сайт дещо відрізняються між собою. Їхня різниця полягає у реалізації багатофункціональності та динамічного оновлення ресурсу [1].

Веб-система акумулює спеціалізовані сторінки. Перша сторінка, на яку заходить користувач, є статичною, її наповнення залишається незмінним, інші сторінки системи потребують інтерактивного інтерфейсу й динамічного оновлення[2].

Шкільний сайт є не тільки візитною карткою навчального закладу. Сучасний веб-ресурс не тільки дозволяє ділитися новинами, він повинен мати й інший розвинений функціонал, орієнтований на підтримку навчального процесу.

Розробка веб-системи навчального закладу забезпечить інформаційний зв'язок вчителів і учнів, надасть віддалений доступ до навчальних матеріалів, дозволить проводити моніторинг навчально-виховного процесу закладу освіти. Потреба в розробці сучасних веб-систем для закладів освіти зростає. Тому актуальною є розробка веб-системи моніторингу та підтримки навчально-виховного процесу у закладах освіти I ступеня.

Зв'язок роботи з науковими програмами, планами, темами. Робота виконувалася згідно плану виконання наукових досліджень на кафедрі програмного забезпечення

Мета та завдання дослідження. Метою бакалаврської дипломної роботи є підвищення ефективності роботи закладу освіти I ступеня шляхом розробки спеціалізованої веб-системи з засобами моніторингу та підтримки навчально-

виховного процесу в молодшій школі, що забезпечить ефективну взаємодію суб'єктів освітнього процесу.

Основними **задачами** роботи є:

- розробити метод моніторингу та підтримки навчально-виховного процесу у закладах освіти з використанням спеціалізованої веб-системи;
- розробити модель роботи веб-системи;
- проаналізувати інформаційне забезпечення сайту;
- розробити програмні засоби реалізації веб-системи моніторингу та підтримки навчально-виховного процесу;
- розробити зручний та зрозумілий графічний інтерфейс веб-ресурсу;
- провести тестування веб-системи відповідно до поставлених задач.

Об'єкт дослідження – процес розробки веб-системи моніторингу та підтримки навчально-виховного процесу у закладах освіти I ступеня.

Предмет дослідження – засоби реалізації веб-системи закладу освіти I ступеня.

Методи дослідження. У процесі дослідження використовувались методи дослідження:

- методи аналізу та синтезу для побудови ключових моделей веб-системи;
- методи побудови компонентів системи та їхніх взаємозв'язків для створення загальної структури збереження та обробки інформаційних ресурсів;
- методи прикладної теорії інформації і теорії алгоритмів для розробки алгоритмів і програмного забезпечення.
- методи тестування для перевірки працездатності системи.

Наукова новизна отриманих результатів.

- Подальшого розвитку отримав метод моніторингу та підтримки навчально-виховного процесу у закладах освіти, який, на відміну від існуючих, базується на використанні комплексного функціоналу моніторингу успішності учнів, наповнення і використання навчально-методичних матеріалів і підтримки виховного процесу, що дозволяє

підвищити ефективність освітнього процесу в умовах дистанційного доступу.

- Подальшого розвитку отримала модель роботи веб-системи закладу освіти I ступеня, яка, на відміну від існуючих, орієнтована на покращенні навчального-виховного процесу шляхом реалізації комплексного функціоналу та розвиненої системи зв'язків для забезпечення взаємодії суб'єктів освітнього процесу, що дозволяє підвищити якість навчання у молодшій школі.

Практична цінність отриманих результатів. Практична цінність полягає у кінцевій реалізації інформаційної веб-системи навчального закладу I ступеня, що дозволить ефективно підтримувати навчально-виховний процес у закладі освіти.

Особистий внесок здобувача. Усі наукові результати, викладені у бакалаврській дипломній роботі, отримані автором особисто. У науковій роботі [3], опублікованій у співавторстві, автору належить розробка модуля електронного журналу та алгоритму моніторингу і підтримки навчально-виховного процесу в школі.

Апробація матеріалів бакалаврської дипломної роботи. Основні положення бакалаврської дипломної роботи доповідалися та обговорювалися на LI науково-технічній конференції факультету інформаційних технологій та комп'ютерної інженерії (2022) ВНТУ.

Публікації. Основні результати дослідження опубліковані в науковій роботі [3] – в тезах доповіді на LI науково-технічній конференції факультету інформаційних технологій та комп'ютерної інженерії (2022) ВНТУ.

Аналіз. У пояснювальній записці до бакалаврської дипломної роботи розглянуто 4 розділи та використано 27 літературних джерел.

1 ОБҐРУНТУВАННЯ ВИБОРУ МЕТОДУ РОЗРОБКИ ТА ПОСТАНОВКА ЗАДАЧ ДОСЛІДЖЕННЯ

1.1 Аналіз стану веб-систем для моніторингу та підтримки навчально-виховного процесу у закладах освіти

У наш час введення сучасних технологій у сферу діяльності людини не можна уявити без використання комп'ютерної техніки. Сьогодні важливо, щоб управління навчальним процесом було ефективним, оскільки воно охоплює широке коло людей: вчителів, учнів, батьків, методистів, впливає на умови їх праці та навчання і має забезпечити конструктивну взаємодію між суб'єктами навчального процесу.

Освіта є важливим сегментом життя кожної людини, адже вона дає знання, навички, різноманітну інформацію. В суспільстві людина має знати свої права та обов'язки. На даний момент в освіті є велика необхідність у використанні сучасних інформаційних технологій.

Кожний освітній заклад повинен мати інформаційну веб-систему для оперативного ознайомлення з новинами та інформацією, де учні та батьки зможуть відслідковувати новини навчального закладу та анонси запланованих подій. Також затребуваною є авторизація на сайті школи, адже ця функція є важлива для тих організацій, яким потрібні дані про відвідувачів. Потрібно зменшити до нуля вірогідність зламування особистого кабінету для збереження інформації про клієнта.

У наш час більшість веб-сайтів використовують авторизацію, адже в окремих випадках авторизація використовується для того, щоб отримати більше можливостей, які недоступні гостям, що відвідали сайт без реєстрації. Наприклад, неавторизовані користувачі не можуть писати коментарі чи ділитися новинами.

Веб-система – це система гіпертекстових документів, зазвичай її ресурси тісно пов'язані між собою. Також можна сказати, що веб-система – це сайт з

динамічними веб-сторінками, які мають відповідати запиту користувача та приваблювати його зрозумілим інтерфейсом.

Головна сторінка, на яку потрапляє користувач, є статичною, оскільки її наповнення завжди незмінне. Статичні сторінки написані мовою розмітки HTML.

Інші сторінки, на які переходить відвідувач через панель меню є динамічними, оскільки формування сторінки відбувається за запитом відвідувача. Динамічний вміст сторінки веб-сайту може змінюватися, це залежить від кількості параметрів конкретної веб-сторінки.

Веб-сайт – це сукупність декількох сторінок, а веб-сторінки – це цифрові файли, написані мовою гіпертекстової розмітки HTML. Веб-сторінки пов'язані між собою і можуть мати однаковий дизайн та інтерфейс. Веб-сайт може містити будь-яку інформацію, різні файли та документи, також можуть бути наявними зображення чи відео [4].

Оскільки Інтернет розвивається і використовується в усіх сферах, веб-сайти відрізняються структурою, наповненням і функціоналом відповідно до їх призначення. Веб-сайт навчально-виховного закладу є актуальним, оскільки дає нові можливості взаємодії в освітньому процесі, він є корисним для батьків, вихованців, учителів та адміністрації [5].

Не дивлячись на велику кількість різноманітних освітніх ресурсів, найефективнішими є ті, що враховують потреби батьків і учнів і підтримуються на рівні закладу освіти.

Освітній сайт – це комплексна структурно побудована система, що пов'язує між собою веб-сторінки, об'єднані спільною темою, ідеєю та призначенням.

Є декілька груп освітніх веб-сайтів:

- сайти, які спонукають розвиток побудови інноваційних освітніх продуктів;
- сайти, які дають можливість відслідковувати навчально-методичний освітній процес;
- сайти, що можуть забезпечувати якісний та просторий доступ до будь-яких освітніх продуктів.

Проаналізувавши тенденції розвитку шкільних сайтів, відзначимо потребу створення спеціалізованого функціоналу, який, крім інформаційного наповнення, виділяється здатністю підтримувати освітній процес, забезпечуючи допоміжні програмні ресурси для виконання спеціалізованих завдань. Тож веб-система моніторингу та підтримки навчально-виховного процесу у закладах освіти є необхідною для розробки і використання, оскільки полегшить вчителям, учням та батькам процеси комунікації і дозволить відслідковувати новини і досягнення, що забезпечить якісну роботу адміністрації навчального закладу і дозволить користувачам системи вчасно отримувати потрібну інформацію.

1.2 Порівняльний аналіз аналогів

На даний момент існує велика кількість закладів освіти, які мають власну веб-систему.

Оскільки інформаційні технології швидко розвиваються, то наявні веб-ресурси потребують постійної підтримки й оновлення, що надає їм більшої функціональності та дозволяє розширити висвітлення інформації про діяльність закладу і забезпечує комутативний зв'язок суб'єктів навчального процесу.

Різноманітні веб-сайти можуть мати різну структуру – кількість веб-сторінок та їх види, внутрішньо-тематичну організацію, сукупність внутрішніх зв'язків. Загальним для всіх веб-сайтів є присутність у структурі головної сторінки, яка пов'язана з цілими тематичними розділами сайту.

Розглянемо декілька варіантів веб-сайтів шкіл:

- КЗ "ЗЗСО І-ІІІ ст. с. Зозулинці" Уланівської сільської ради;
- Лукашівський навчально - виховний комплекс;
- Лавриківський заклад загальної середньої освіти І-ІІІ ступенів;
- ЗЗСО І-ІІІ ступенів ім. Г.Т. Танцюри с. Зятківці.

Веб-сайт комунального закладу "Заклад загальної середньої освіти I-III ступенів с. Зозулинці" (рис. 1.1) є доволі професійним, але недоліки цього веб-сервісу можна побачити неозброєним оком [6].

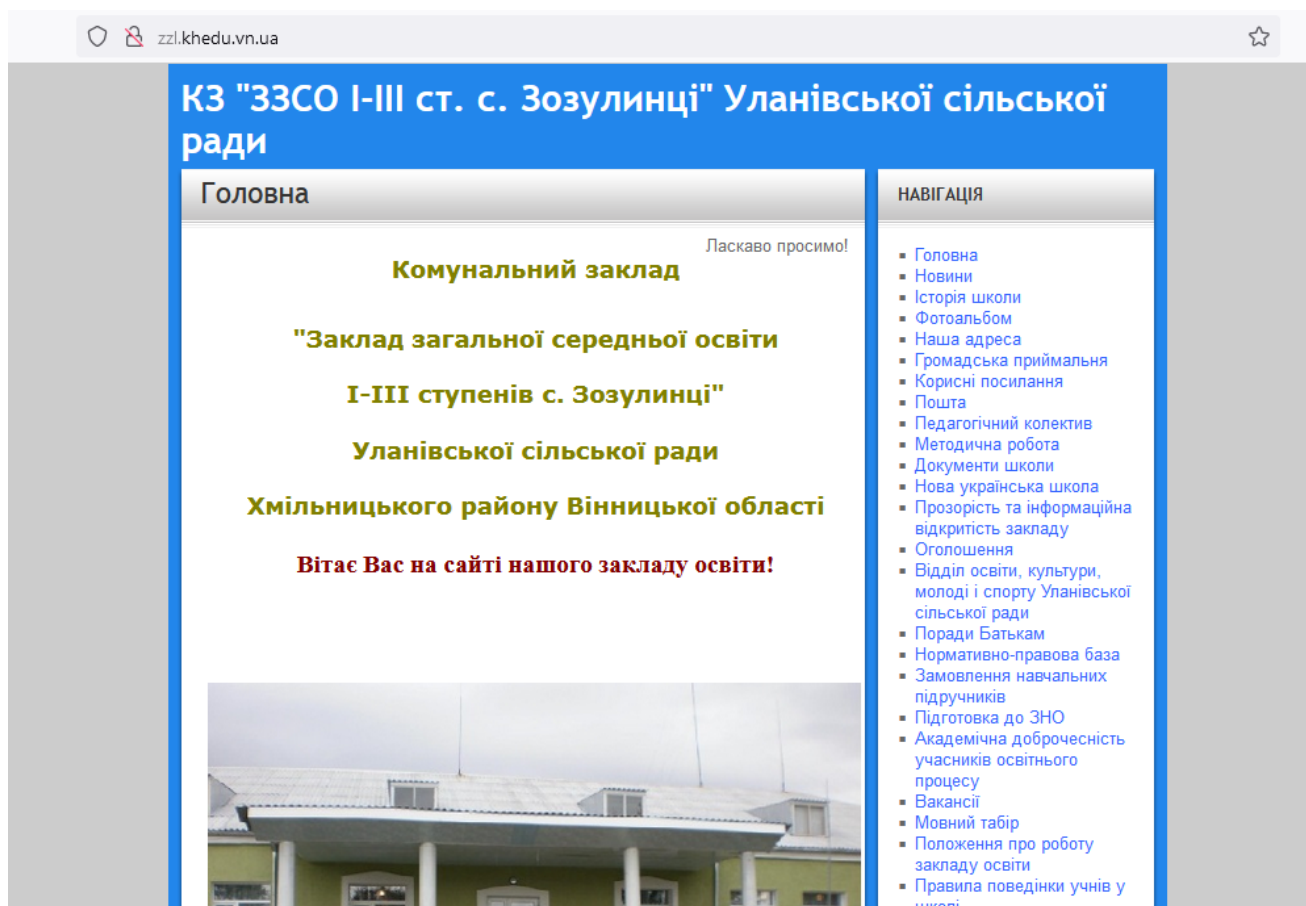


Рисунок 1.1 – Головна сторінка сайту комунального закладу "Заклад загальної середньої освіти I-III ступенів с. Зозулинці"

Проаналізувавши даний сайт, можна сказати, що сайт має застарілий дизайн, навігація має багато пунктів, деякі з них є незрозумілими, а також є деякі пункти меню, перейшовши за якими можна побачити порожні сторінки.

Веб-система не має пошуку по сайту, а також у веб-сайті не можна увійти в кабінет для батьків, де можна відслідковувати оцінки своїх дітей.

Можна припустити, що даний сайт був створений за допомогою конструктора.

Веб-сайт Лукашівського навчально-виховного комплексу (рис.1.2) має приємну кольорову гаму, зрозуміле меню, де є все необхідне для візуалізації матеріалів по підтримці навчально-виховного процесу [7].

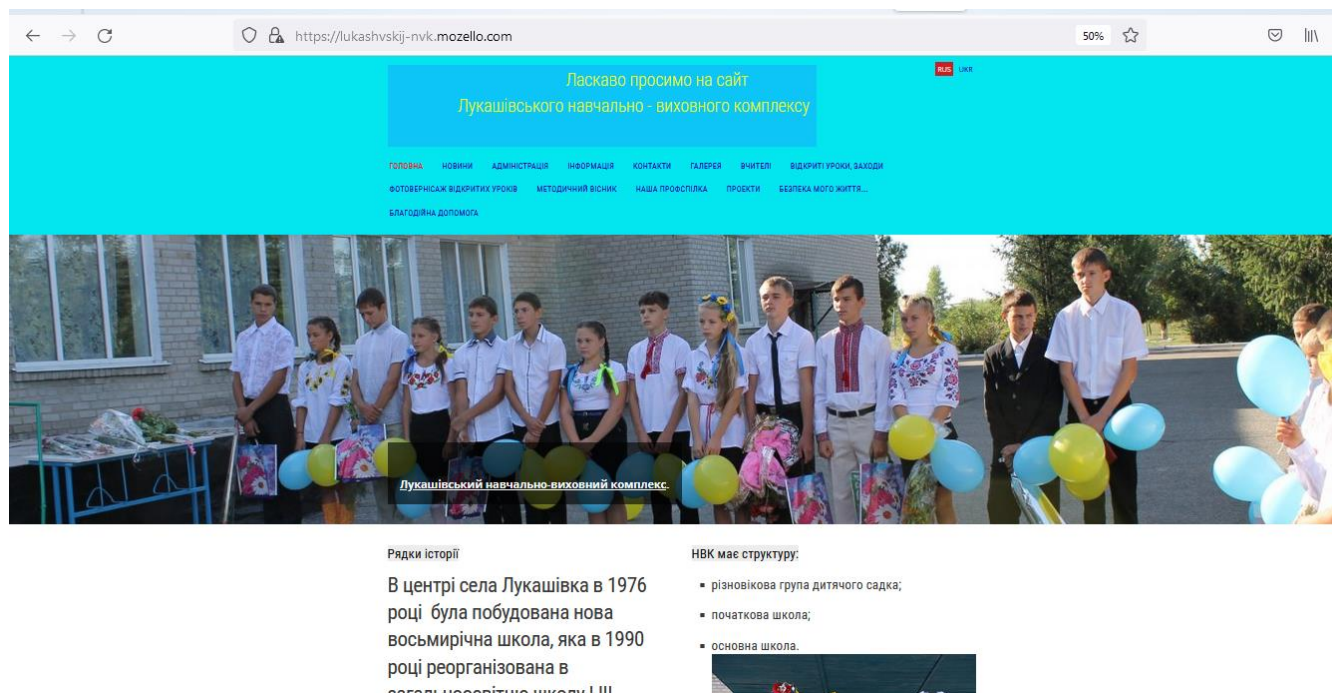


Рисунок 1.2 – Головна сторінка сайту Лукашівського навчально – виховного комплексу

Аналізуючи даний сайт, також були знайдені і мінуси. Шапка сайту розтягнута на всю сторінку, а текст на головній сторінці займає надто малу частину на сайті. Текст має різний шрифт і розмір. На сайті є можливість змінити мову. Тут можна сміливо стверджувати, що даний сайт було розроблено за допомогою автоматизованої системи створення сайтів, яких зараз є безліч.

Веб-сайт Лавриківського закладу загальної середньої освіти І-ІІІ ступенів (рис 1.3) було подано в Google конструкторі. Сайт має пошук по веб-сторінках, а також зрозумілу навігацію [8].

Проаналізувавши даний сайт, стало зрозуміло, що кольорова гама є надто темною, а темні кольори не приносять дітям радість і навпаки можуть їх засмутити, а також є неприємними для використання в дизайнерському

оформленні сайту. Дизайн цього сайту є застарілим. Також ще одним недоліком, як і в інших сайтах, є те, що на сайт не можна зайти особисто під своїм паролем та логіном.

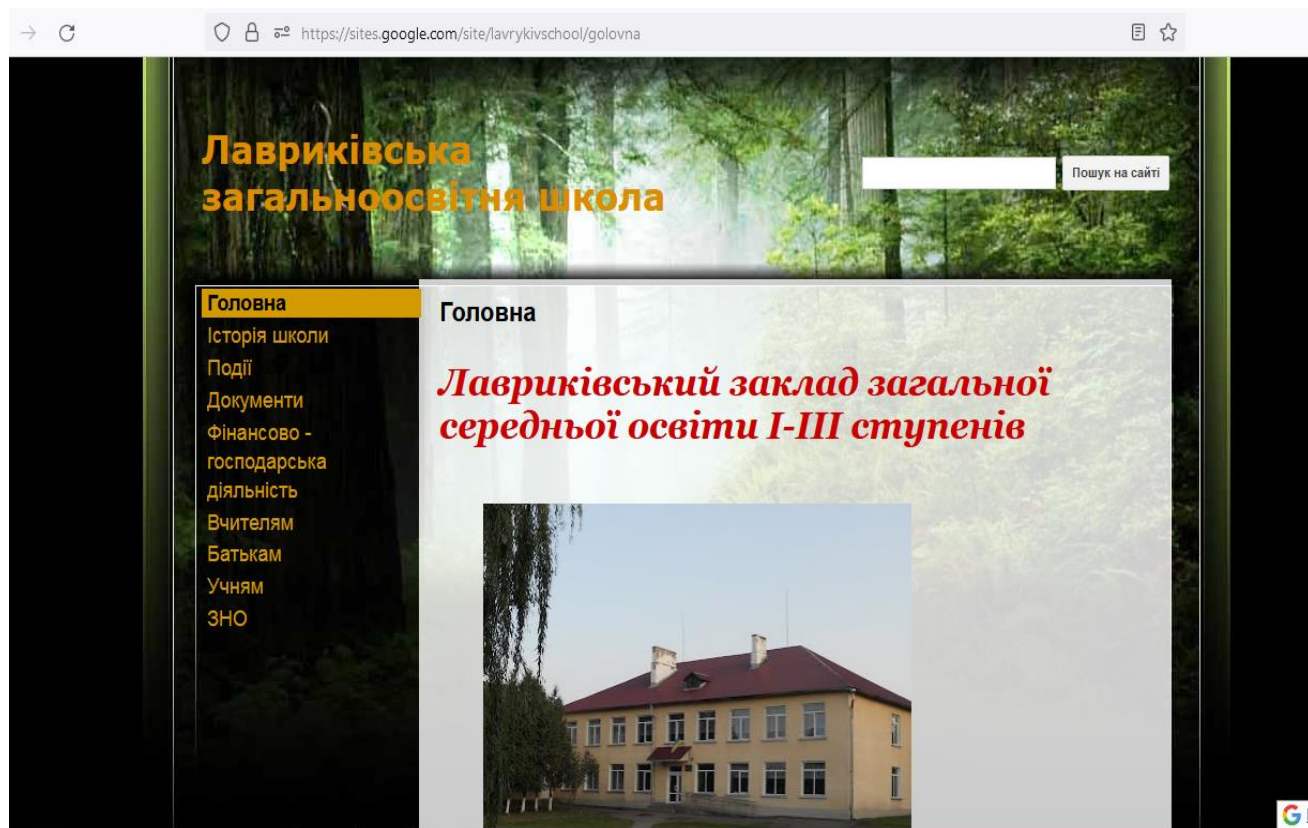


Рисунок 1.3 – Головна сторінка сайту Лавриківського закладу загальної середньої освіти I-III ступенів

Протестувавши сайт ЗЗСО I-III ступенів ім. Г.Т. Танцюри с. Зятківці [9] було виявлено як переваги та недоліки даного сайту.

Сайт має широку навігаційну панель, також є гіперпосилання, за допомогою яких можна завжди повернутися на головну сторінку, кольорова гамма є звичайною, відсутня шапка сайту та логотип.

Також сайт було протестовано у різних веб-браузерах і стало зрозумілим те, що сайт не адаптований під різні браузери.

Перейшовши на інші сторінки веб-сайту, було виявлено їх порожніми, можливо даний сайт знаходиться ще в розробці.

Також на сайті немає можливості авторизуватися. Головну сторінку сайту наведено на рисунку 1.4.



Рисунок 1.4 – Головна сторінка сайту ЗЗСО I-III ступенів ім. Г.Т. Танцюри с. Зятківці

Результати порівняння аналогів зведено в табл. 1.1.

Таблиця 1.1 – Порівняльна характеристика аналогів

Критерії	Сайт Лавриківського закладу загальної середньої освіти	Сайт Лукашівського навчально – виховного комплексу	Сайт Закладу загальної середньої освіти с. Зозулинці	Сайт ЗЗСО I-III ступенів ім. Г.Т. Танцюри с. Зятківці	Власна веб-система
Зручна навігація	-	+	+	+	+

Продовження таблиці 1.1

Авторизація	-	-	-	-	+
Новини	+	+	+	+	+
Читабельність тексту	+	-	+	+	+
Зрозуміле меню	+	+	+	+	+
Загальна оцінка	60%	60%	80%	80%	100%

Таблиця порівняльних характеристик показала, що розробка власної веб-системи є доцільною. В результаті отримаємо веб-сайт, який покриває недоліки існуючих рішень.

1.3 Аналіз методів розв'язання задачі

Для розв'язання задачі створення веб-системи моніторингу та підтримки навчально-виховного процесу у закладах освіти I ступеня існує декілька варіантів рішень. Але, якщо розібратися в цілому, то зрозуміло, що немає однієї стандартної класифікації сайтів. Класифікацію веб-сайтів можна поділити за призначенням:

- персональна сторінка – це сайти, які раніше могли складатися лише з однієї сторінки, але зараз до цього типу належать усі багатосторінкові ресурси, такі як блоги;
- сайт-візитка – це сайт, який може належати одному власнику або невеликій групі осіб, такий сайт може містити декілька сторінок, він найкраще підходить для малого бізнесу, де розповсюджують свою рекламу власних послуг;
- інформаційний сайт – це сайт, який має велику кількість інформації, де є багато тематичних розділів, тобто такий веб-ресурс створюється для

користувача як основне джерело інформації, таким сайтом може бути, наприклад, бібліотека, де знаходиться велика кількість різноманітних матеріалів;

- корпоративно-інформаційний сайт – це сайт, на якому розміщено детальну інформацію та який є необхідними для компанії, організації, школи, ВНЗ тощо; якщо розглядати такий сайт для школи, то в ньому може бути розміщена інформація про історію школи, події, новини, шкільне життя і т.д. [10].

Відомо, що всі сайти, які знаходяться в мережі Інтернет, можна розділити на статичні та динамічні. Зазвичай відвідувачу не важливо, на якій сторінці він перебуває, інколи він не розуміє статичною чи динамічною є сторінка.

Статичний сайт – це сайт, який складається із статичних сторінок, які написані мовою розмітки. Такі веб-сторінки завжди цілісно можуть зберігатися на сервері і бути візуально незмінними. Такі сайти можуть містити фото, текст, відео, але у вигляді HTML розмітки. Також статичні сайти дають зовсім незначне навантаження на сервер і їх легко перенести на нього. Щоб внести будь-які зміни в сайт потрібно вносити їх у вихідний код документу такого веб-ресурсу [11]. Приклад статичного сайту сайту-візитки зображено на рисунку 1.5.

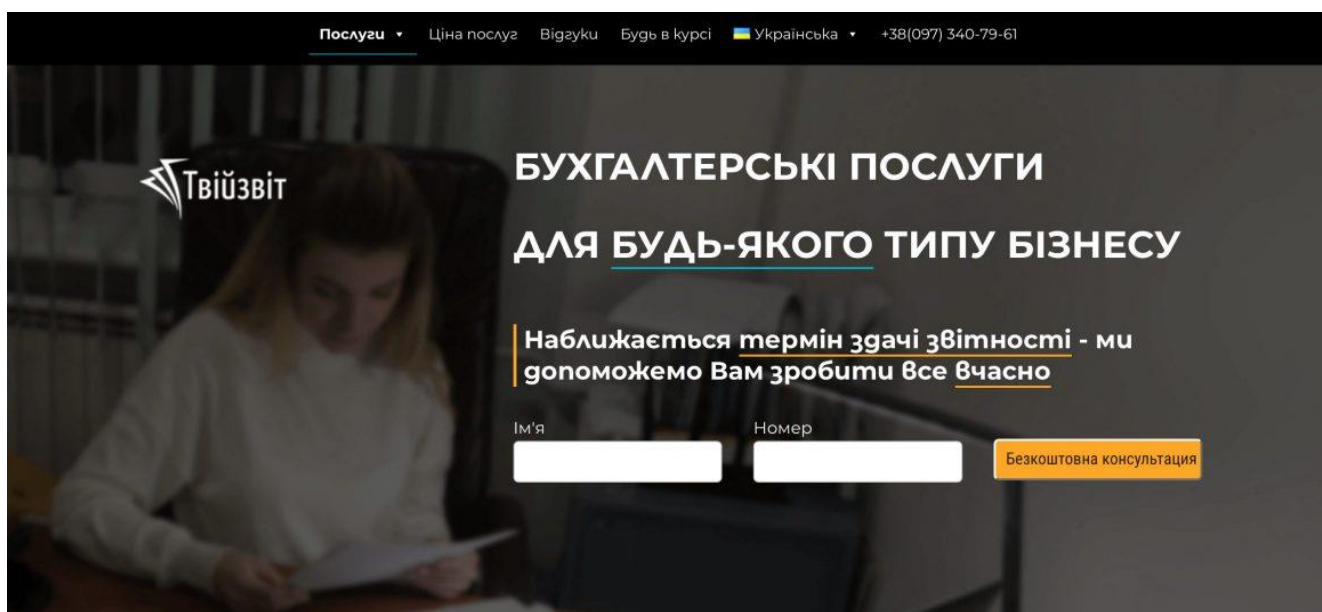


Рисунок 1.5 – Приклад статичного сайту-візитки

Динамічний сайт – сайт, котрий складається зі сторінок, які можуть змінюватися (рис. 1.6). Зазвичай такі сторінки використовують, щоб показати ціни на акції, чи інформацію, яка потребує постійного оновлення, наприклад, дані про погоду – це приклади, де інформація на веб-ресурсі змінюється в динамічному режимі. Код такого типу сайту генерується під час обробки HTML файлу інтерпретатором обраної мови програмування. Перевагою такого виду сайту є можливість самостійно, без будь-якої допомоги вносити зміни на сайті, а ще одним із плюсів є швидке відображення нововведених змін. Недоліком такого сайту є високе навантаження на сервер. Мови програмування тут можуть використовуватися різні, але більш поширеними у використанні є Java та PHP. При розробці динамічного типу сторінки зазвичай шаблон дизайну зберігається, такий варіант є досить зручним [12].

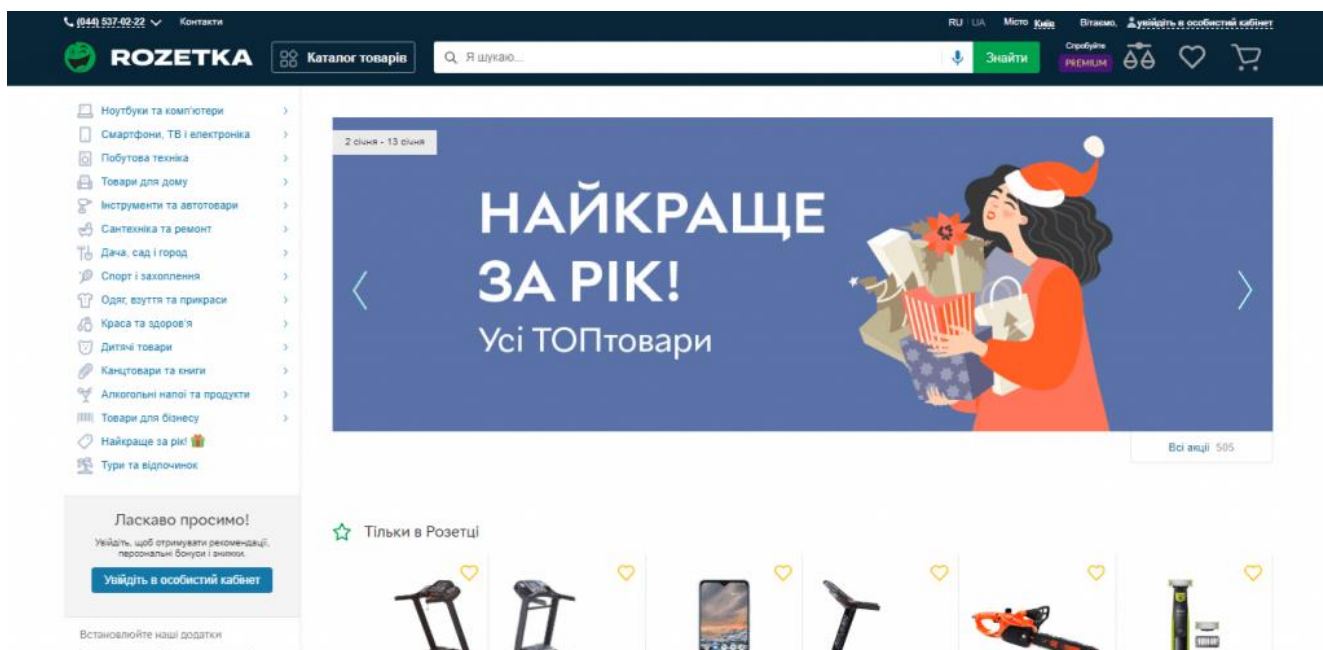


Рисунок 1.6 – Приклад динамічного сайту

Отже, проаналізувавши усі переваги та недоліки різних типів веб-ресурсів, одним із основних методів розв'язування задачі створення веб-системи навчального закладу зі структурою корпоративно-інформаційного сайту є

розробка веб-ресурсу з динамічним наповненням сторінок, оскільки динамічний сайт надає більше можливостей для користувачів сайту такого призначення.

1.4 Постановка задач для веб-системи моніторингу та підтримки навчально-виховного процесу

Проаналізувавши переваги і недоліки розглянутих сайтів закладів освіти, було обґрунтовано актуальність розробки веб-системи моніторингу та підтримки навчально-виховного процесу і сформульовано задачі бакалаврської дипломної роботи:

- розробити метод моніторингу та підтримки навчально-виховного процесу у закладах освіти з використанням спеціалізованої веб-системи;
- розробити модель роботи веб-системи;
- проаналізувати інформаційне забезпечення сайту;
- розробити програмні засоби реалізації веб-системи моніторингу та підтримки навчально-виховного процесу;
- розробити зручний та зрозумілий графічний інтерфейс веб-ресурсу;
- провести тестування веб-системи відповідно до поставлених задач.

1.5 Висновки

У першому розділі було розглянуто перспективи розвитку веб-ресурсів закладів освіти та обґрунтовано доцільність розробки веб-системи для моніторингу і підтримки навчально-виховного процесу у закладах освіти I ступеня. Було проведено порівняльний аналіз аналогів веб-систем шкіл та навчально-виховних комплексів, основним призначенням яких є інформування відвідувачів про події і різноманітні новини, які стосуються школи. Серед аналогів розглянуто сайти КЗ "ЗЗСО I-III ст. с. Зозулинці" Уланівської сільської ради, Лукашівського навчально-виховного комплексу, Лавриківського закладу загальної середньої освіти I-III ступенів та ЗЗСО I-III ступенів ім. Г.Т. Танцюри с. Зятківці.

Проаналізувавши плюси та мінуси кожного із веб-сайтів, було прийнято рішення розробити власну веб-систему для моніторингу та підтримки навчально-виховного процесу зі зручним інтерфейсом, зрозумілою навігацією, з можливістю авторизуватися користувачу для отримання доступу до конференційної інформації індивідуального призначення. Було сформульовано задачі, які потрібно виконати в бакалаврській дипломній роботі.

Розробка веб-ресурсу проводиться на замовлення Малинківського НВК.

2 РОЗРОБКА МЕТОДУ, МОДЕЛІ ТА АЛГОРИТМІВ РОБОТИ ВЕБ-СИСТЕМИ

2.1 Аналіз інформаційного забезпечення веб-системи

Малинківський НВК – це навчально-виховний комплекс, який знаходиться у Погребищенському районі Вінницької області. Навчально-виховний комплекс охоплює біля 40 дітей. На території школи знаходиться два дитячі майданчики для дітей різного віку.

Головним завданням при написанні веб-сайту є визначення інформаційного наповнення системи, зокрема, внесення інформації про публікації, новини, події, а також ведення електронного журналу з оцінками учнів для моніторингу їх успішності. Важливо розуміти, що одним з основних етапів розробки сайту є саме грамотне інформаційне забезпечення ресурсу з динамічним оновленням матеріалів.

На головній сторінці веб-сайту потрібно надати необхідну інформацію користувачам, зокрема, батькам, учням, вчителям й адміністрації, а також гостям сайту. На головній сторінці має бути присутнє фото школи та її девіз. Також важливо надати інформацію про історію школи та динамічно оновлювати актуальні новини й оголошення, що інформують користувачів про проведені й заплановані заходи. Перейшовши на іншу сторінку, завжди має бути можливість повернутися на головну за допомогою однієї дії, це натискання на назву сайту, вона завжди надасть можливість отримати доступ до головної сторінки веб-сайту.

Сторінка «Про школу» міститиме інформацію про школу, її історію, а також про її працівників. Інформація має бути надана у вигляді тексту, який повинен бути читабельним і зручним для сприйняття відвідувачами сайту. Веб-система повинна мати приємну колірну гаму інтерфейсу.

Сторінка «Галерея» міститиме фото та відео з життя учнів школи на різних подіях та заходах.

На сторінках «Батькам» та «Учням» буде розміщена актуальна інформація для батьків та учнів, а також для дітей міститимуться посилання на цікаві та

розвиваючі ігри. Це надасть учням більше користі від відвідування сайту, а також підвищить бажання відвідувати сайт школи.

Веб-сторінка «Життя школи» буде висвітлювати інформацію про події, які відбуваються у навчально-виховному закладі, зокрема, міститиме фото й описи різних подій.

Також при авторизації з'являтиметься сторінка «Журнал», де вчителі можуть побачити інформацію про учнів школи та їх батьків, а також виставити бали учнів в електронний журнал з різних предметів.

Кольорова гама буде реалізована в легко блакитних кольорах, які не ріжуть очі і є приємними для користувача.

2.2 Розробка структури інтерфейсу веб-системи

Правильна послідовність етапів проєктування системи – гарантія успіху кінцевого продукту. На перший погляд може здатися, що тут складного – продумав дизайн, написав код, наповнив систему текстом – і готово. Але насправді кожен етап створення сайту – це важлива робота, за яку несе відповідальність розробник. Створення веб-системи охоплює не просто розробку дизайну і написання коду, а й розгорнутий аналіз проєкту.

Привабливий та зручний інтерфейс є важливим показником якості будь-якого сайту. Всі елементи інтерфейсу мають бути взаємопов'язаними та логічно структурованими. Головним завданням веб-інтерфейсу є потреба зробити перебування на сайті максимально простим та зрозумілим користувачу.

Особливості розробки інтерфейсу сайту полягають в інформаційній структурі, плануванні взаємодії та розумній привабливості ресурсу. Зрозуміла та грамотна структура сайту спростить життя користувачу, адже тоді відвідувач веб-сайту досягне того результату, який потрібен, і витратить мінімум сил і часу для отримання шуканої інформації [13].

Чим більш просту структуру та інтерфейс бачить користувач, тим легше йому зорієнтуватися в навігації й тим менше часу він тратить на те, щоб знайти

потрібну інформацію, оскільки не буде витрачати час на адаптацію до нового інтерфейсу й почуватиметься більш впевнено [14].

При розробці інтерфейсу було прийнято рішення зробити його простим у використанні, орієнтованим на не підготовленого користувача.

При розробці веб-системи було створено головне меню (рис. 2.1), де відвідувач може відкривати сторінку для відображення. Користувач також може скористатися входом в особистий кабінет, щоб отримати більший доступ до ресурсів сайту.

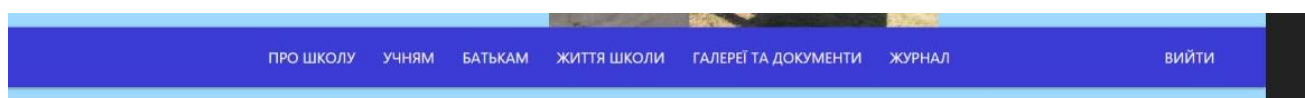


Рисунок 2.1 – Головне меню

Для зручного доступу до електронного журналу вчителям надана можливість авторизації на веб-сторінці (рис. 2.2). Кожний вчитель, щоб потрапити на сторінку свого класу із журналом, використовує власний логін і пароль.

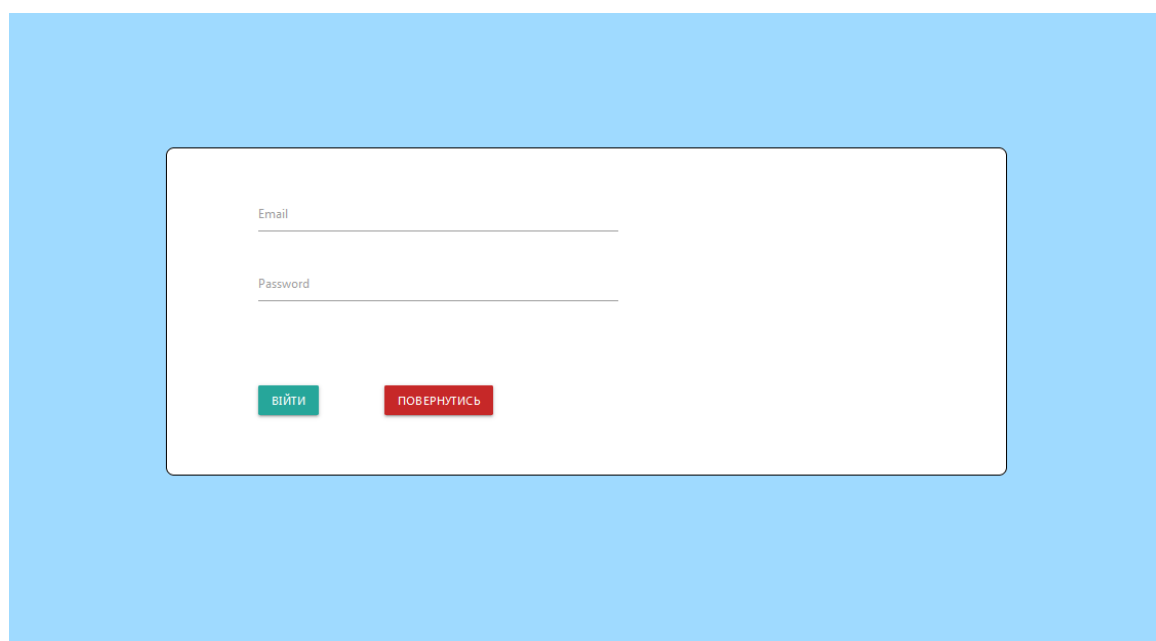


Рисунок 2.2 – Вікно авторизації

Було розроблено логотип для шапки сайту, що є важливим елементом графічного дизайну та фірмового стилю. Логотип є образотворчим візитним знаком з цифрами, літерами, чи без них, що є емблемою конкретної організації. Логотип завжди виражає індивідуальність сайту, тому, чим креативнішим є логотип, тим більша його репрезентативність (рис 2.3). Логотип акумулює естетичність, характерність, простоту та індивідуальність, все це тісно пов'язано між собою.



Рисунок 2.3 – Логотип для шапки сайту

Розробка графічного інтерфейсу веб-системи була проведена у середовищі розробки Microsoft Visual Studio Code.

2.3 Розробка методу моніторингу та підтримки навчально-виховного процесу

Одним із основних питань освітнього процесу є постійний контроль за рівнем якості навчання. Оскільки без спеціального відстеження цього процесу неможливо оцінити якість освіти, було розроблено метод моніторингу та підтримки навчально-виховного процесу у закладах освіти I ступеня.

Запропонований метод передбачає, що у вчителя є можливість мати доступ до електронного журналу, в якому акумульована інформація про учнів, їх місце проживання, контакти батьків, а також до сторінок журналу з оцінками по предметах, в яких кожен з педагогів має виставляти оцінки учням. Така реалізація дозволяє адміністрації школи, вчителям і батькам моніторити поточну успішність кожного учня. Щоб отримати доступ до функціоналу журналу, з метою захисту даних та для захисту персональної інформації, користувачам веб-системи потрібно пройти авторизацію.

Розроблений метод моніторингу та підтримки навчально-виховного процесу в закладах освіти I ступеня включає послідовність дій:

1. Авторизація користувачів у веб-системі.
2. Перехід в особистий кабінет.
3. Перегляд в електронному журналі інформації про учнів.
4. Внесення вчителями змін у журнал щодо інформації про учнів, батьків, навчальні предмети і їх базові параметри.
5. Внесення поточних і підсумкових оцінок.
6. Перевірка й корегування інформації.
7. Моніторинг успішності конкретного учня чи класу за вказаний період.

Схема реалізації методу моніторингу та підтримки навчально-виховного процесу в закладах освіти наведена на рис. 2.4.



Рисунок 2.4 – Схема реалізації методу моніторингу та підтримки навчально-виховного процесу в закладах освіти

2.4 Розробка моделі роботи веб-системи

Модель – це формалізоване подання групи певних правил та даних, які надалі будуть застосовуватися для роботи з системою і визначати основу управління її функціонуванням.

Щоб зрозуміти, як працює веб-система для моніторингу та підтримки навчально-виховного комплексу, було розроблено модель роботи системи із застосуванням UML діаграми (рис. 2.5). Функціонал діаграми дозволяє описати програму усіх даних процесів [15]. Діаграма акумулює базові модулі системи і визначає множину взаємозв'язків між ними. Зокрема, щоб увійти на сторінку журналу, учитель має авторизуватися, тобто ввести логін і пароль, який до нього прив'язаний. Якщо введення некоректне, система про це повідомляє і дає можливість ввести дані повторно. Після введення коректних даних користувач отримує доступ до визначених ресурсів.

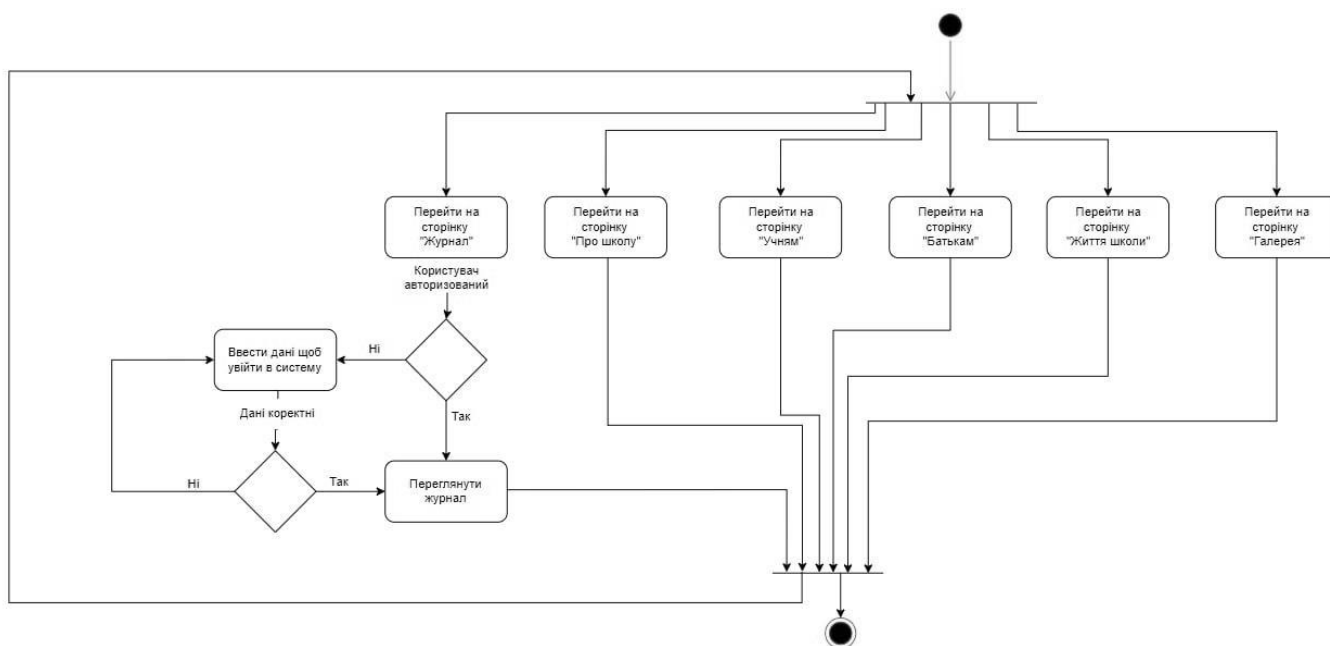


Рисунок 2.5 – Модель роботи веб-системи

2.5 Розробка алгоритмів роботи веб-системи

Для створення веб-системи потрібно розробити загальний алгоритм, за яким працюватиме система (рис. 2.6).

За алгоритмом першим кроком користувач отримує медіа-дані на сайті. Наступні кроки можливість передбачають інтерактивної взаємодії з сайтом.

Реєстрація й авторизація на сайті дає доступ до персоналізованих ресурсів.

Гість сайту може переглядати тільки загальну інформацію.

Залежно від наданих прав авторизований користувач отримує доступ до системи моніторингу навчально-виховного процесу.

Батьки можуть відслідковувати поточну успішність своїх дітей за визначений період.

Адміністрація закладу й вчителі отримують право наповнювати систему й корегувати дані.

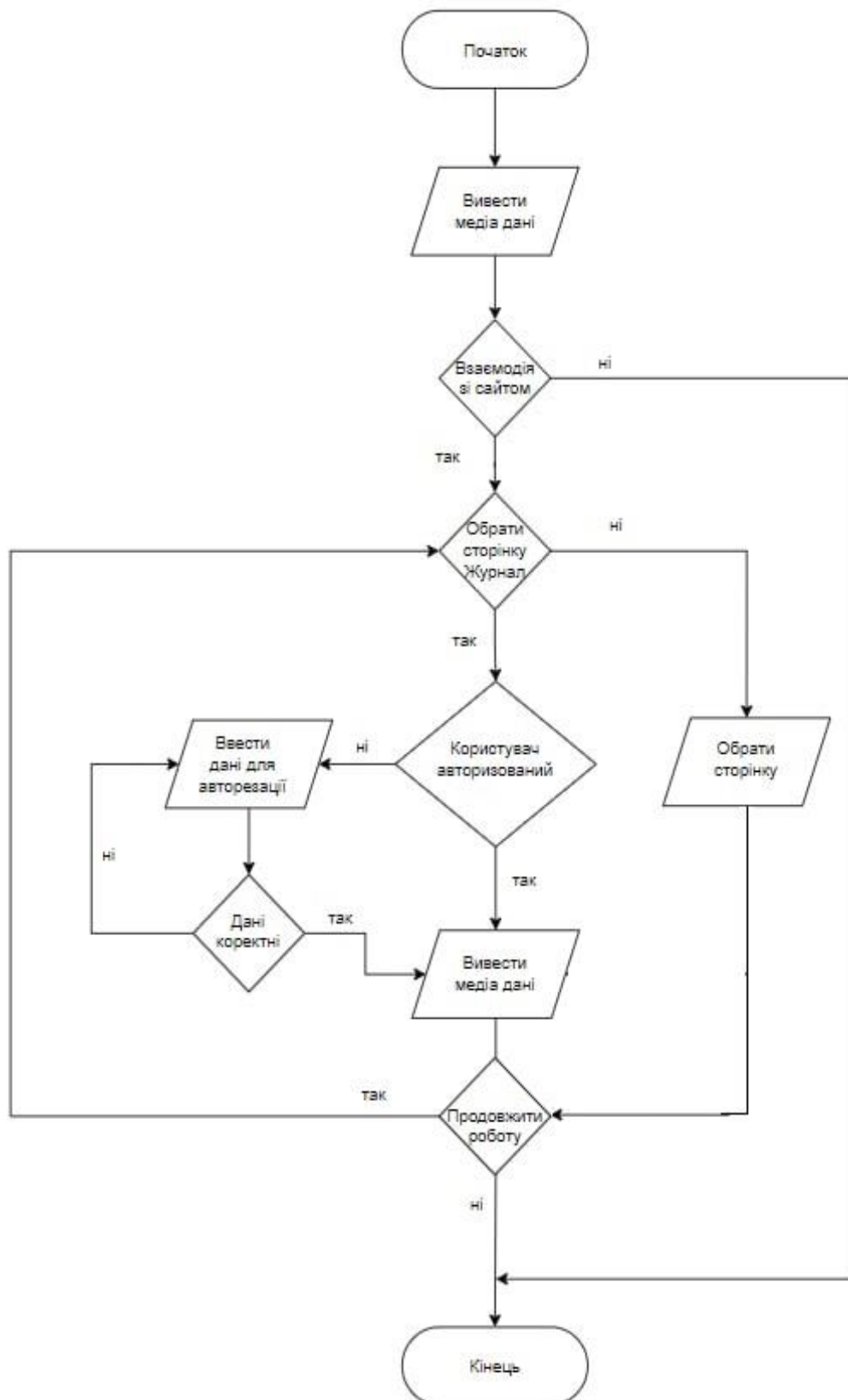


Рисунок 2.6 – Блок-схема загального алгоритму програми

Щоб краще зрозуміти роботу модуля електронного журналу, було розроблено додатковий алгоритм (рис. 2.7- 2.8). Першим етапом алгоритму є

введення ID вчителя. Ці дані дають змогу записати назву предмета. Далі відбувається проходження до введення ПІБ учнів. Наступним кроком є запис оцінок у електронний журнал.

Розроблені алгоритми орієнтовані на автоматизацію процесу наповнення баз даних веб-системи.

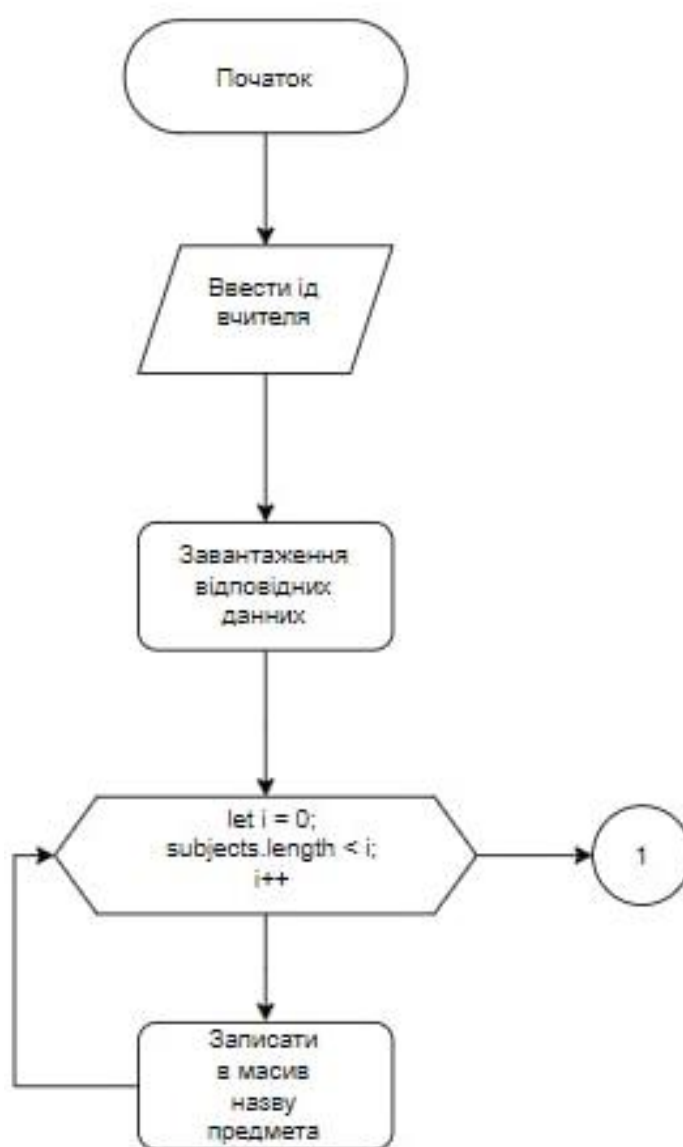


Рисунок 2.7 – Блок-схема алгоритму роботи електронного журналу

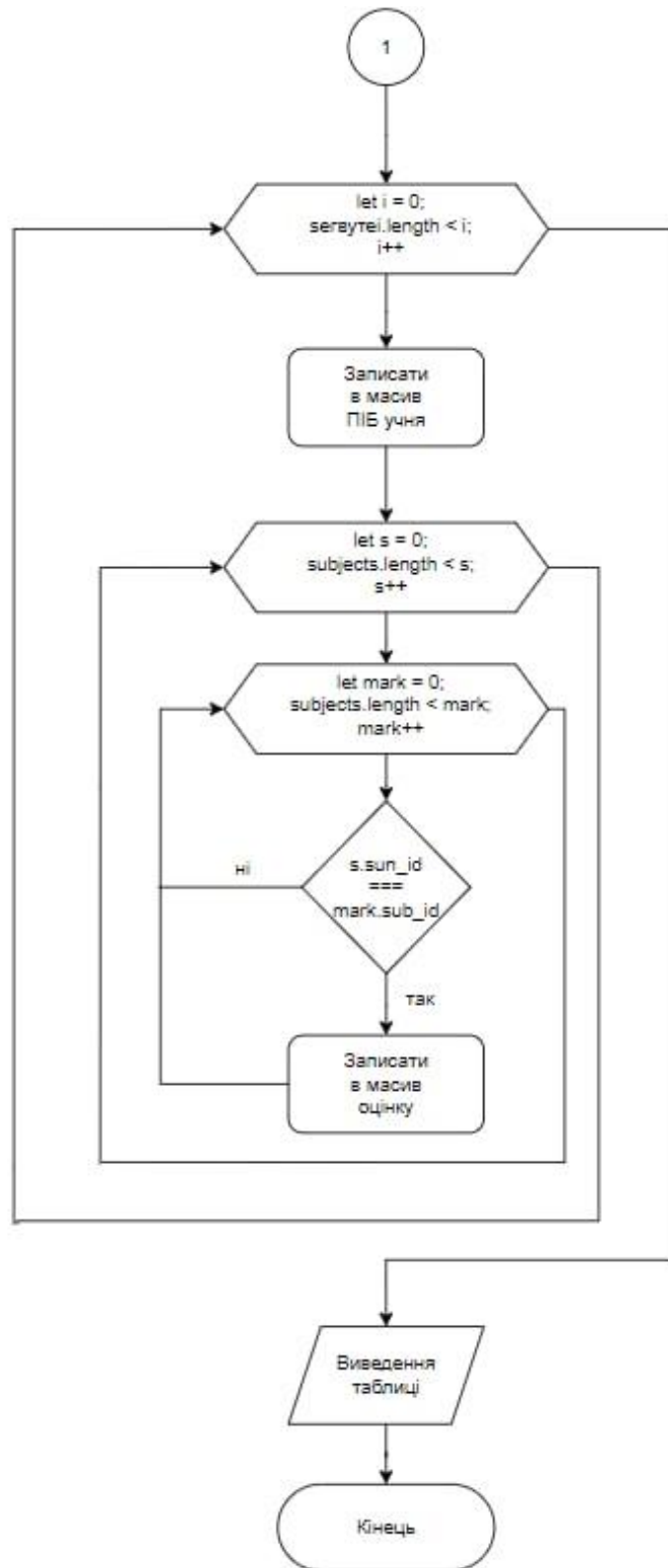


Рисунок 2.8 – Блок-схема алгоритму роботи електронного журналу (продовження)

2.6 Висновки

Отже, у другому розділі було проведено аналіз даних веб-системи навчального закладу I ступеня.

Розроблено структуру інтерфейсу веб-ресурсу, метод моніторингу та підтримки навчально-виховного процесу в закладах освіти, модель і алгоритм роботи веб-системи.

Розроблені засоби дозволяють формалізувати правила роботи системи і забезпечують якісну програмну реалізацію веб-ресурсу.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ ВЕБ-СИСТЕМИ ДЛЯ МОНІТОРИНГУ ТА ПІДТРИМКИ НАВЧАЛЬНО-ВИХОВНОГО ПРОЦЕСУ

3.1 Варіантний аналіз і обґрунтування вибору засобів для реалізації програмного засобу

У наш час існує велика кількість різних засобів для створення веб-сайтів. Є декілька методів для розробки веб-системи [16]:

- самописна – за допомогою однієї або декількох мов програмування;
- автоматизована – за допомогою інструментальних систем (CMS).

Самописна або «ручна» робота може здійснюватися як у простих, так і візуальних редакторах HTML та CSS, такий метод розробки сайту завжди дає гарантію унікальності сайту, оскільки веб-додаток одразу реалізує всі побажання замовника, а також його унікальність наявна у коді, налаштуванні та веб-дизайні. Є можливість вносити будь-які виправлення без жодних обмежень та налаштовувати їх під той результат, який потрібний замовнику [17].

Кожен веб-сайт, побудований з використанням трьох мов програмування — HTML, CSS та JavaScript. JavaScript, виступає «основою» розробки й відповідає за інтерактивність і взаємодію з користувачем.

JavaScript – це мова програмування для створення інтерактивних веб-сторінок, а також мова, яка дозволяє реалізовувати низку рішень веб-документів. Ця мова програмування надає веб-сторінці додаткову динаміку та розширює її можливості.

JavaScript є об'єктно-орієнтовна клієнтська мова, яка підтримується додатками, що працюють із дизайном веб-сайту. Можна сказати, що це третя важлива мова поруч із HTML та CSS, за допомогою якої розробляється велика кількість стандартних веб-інтерфейсів. Ця мова має подібний синтаксис до мови С.

JavaScript має набір певних вбудованих операцій, які не обов'язково можуть бути методами чи функціями, а також набір вбудованих операторів, що можуть

управляти логікою виконання програми. Ця мова програмування має свої плюси, вона може працювати в різних браузерах, для неї не є обов'язковим компілятор, оскільки браузер інтерпретує її в HTML. Основна перевага мови JavaScript – це динамічність, гнучкість роботи та універсальність [18].

Серед відомих веб-редакторів виділяють такі:

- Microsoft FrontPage – це простий у використанні веб-редактор для розробки веб-сайтів (рис 3.1). Він легкий у вивченні, для його опанування вистачить декілька годин. FrontPage надає можливість творити сайти високої складності, оскільки програма надає доступ до великих функціональних можливостей [19].

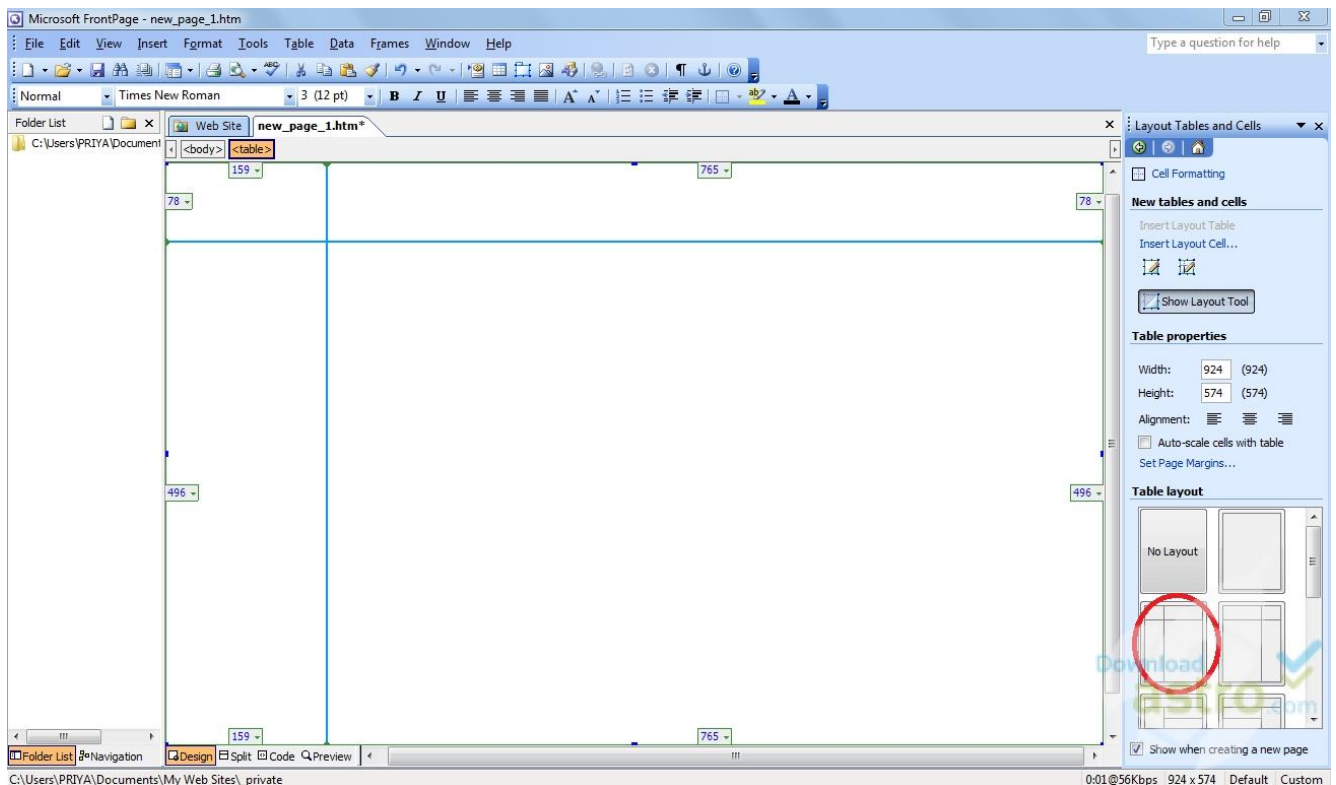


Рисунок 3.1 – Приклад інтерфейсу Microsoft FrontPage

- Visual Studio Code – це легкий редактор коду, який має служби для різних мов програмування, а також потужне середовище для розробки програмних продуктів (рис 3.2). Його можна використовувати не тільки на Windows, але й на Linux. Також цей редактор використовує Node.js. Node.js – це платформа, яка призначена для створення серверних додатків за допомогою

JavaScript, яка легко масштабується, тобто при підключенні великої кількості користувачів до сервера працює асинхронно й розставляє пріоритети грамотно. Щоб запустити цей додаток, на комп'ютер потрібно завантажити середовище Node.js [20].

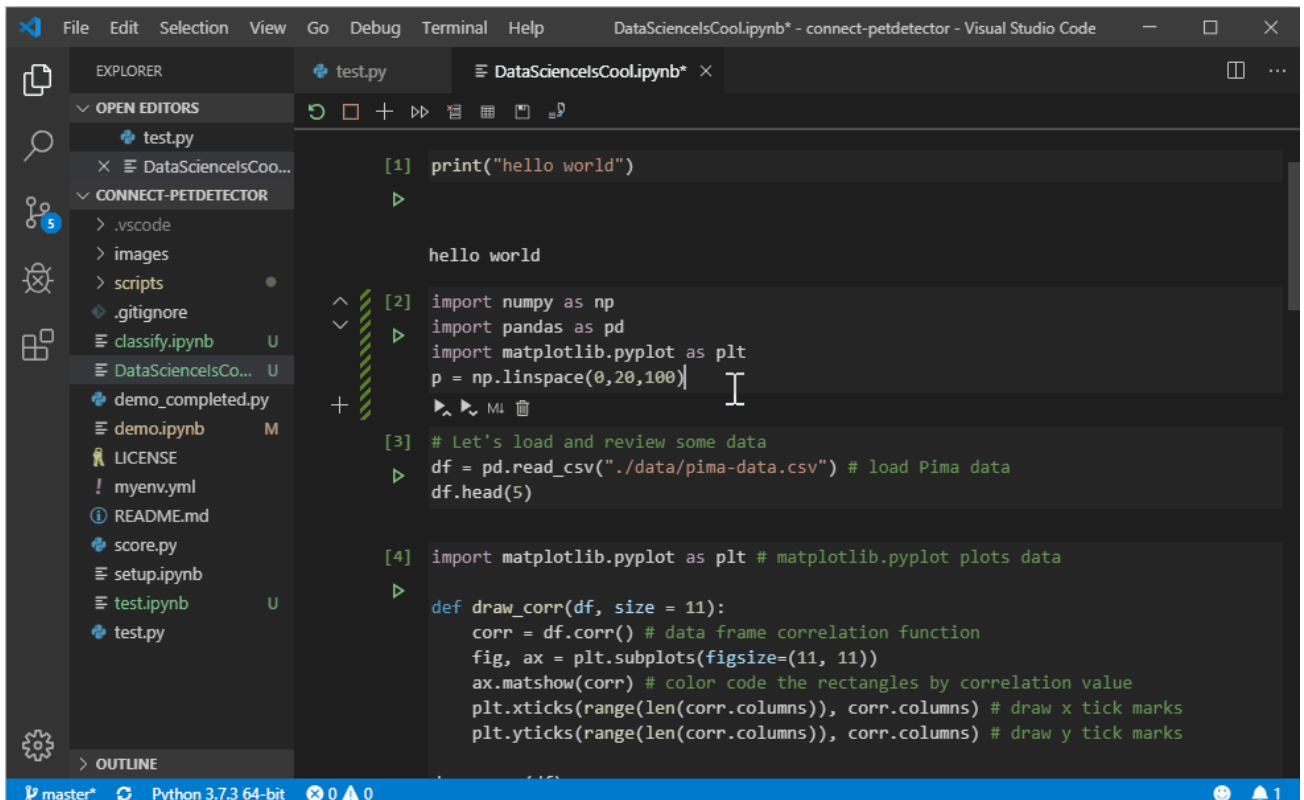


Рисунок 3.2 – Приклад інтерфейсу Visual Studio Code

Наступний метод – це написання сайтів з використанням інструментальних систем. CMS – це готова платформа для створення сайту, так званий конструктор. Даний метод у наш час користується чималою потребою серед фахівців, оскільки забезпечує велику економію часу при розробці. Але є й суттєвий недолік даного засобу. Такі платформи не завжди задовольняють своїм функціоналом і технічною характеристикою [21]. Найбільш розповсюдженими CMS системами є:

- Joomla – це одна із найпростіших та доступних програм, на якій можна розробляти ресурси з просунутим функціоналом. Вона займає друге місце у світі за поширеністю. Як показує практика і опитування, на цьому ресурсі легко і просто створювати веб-сайти [22]. Приклад інтерфейсу програмного додатку Joomla наведено нижче на рисунку 3.3.

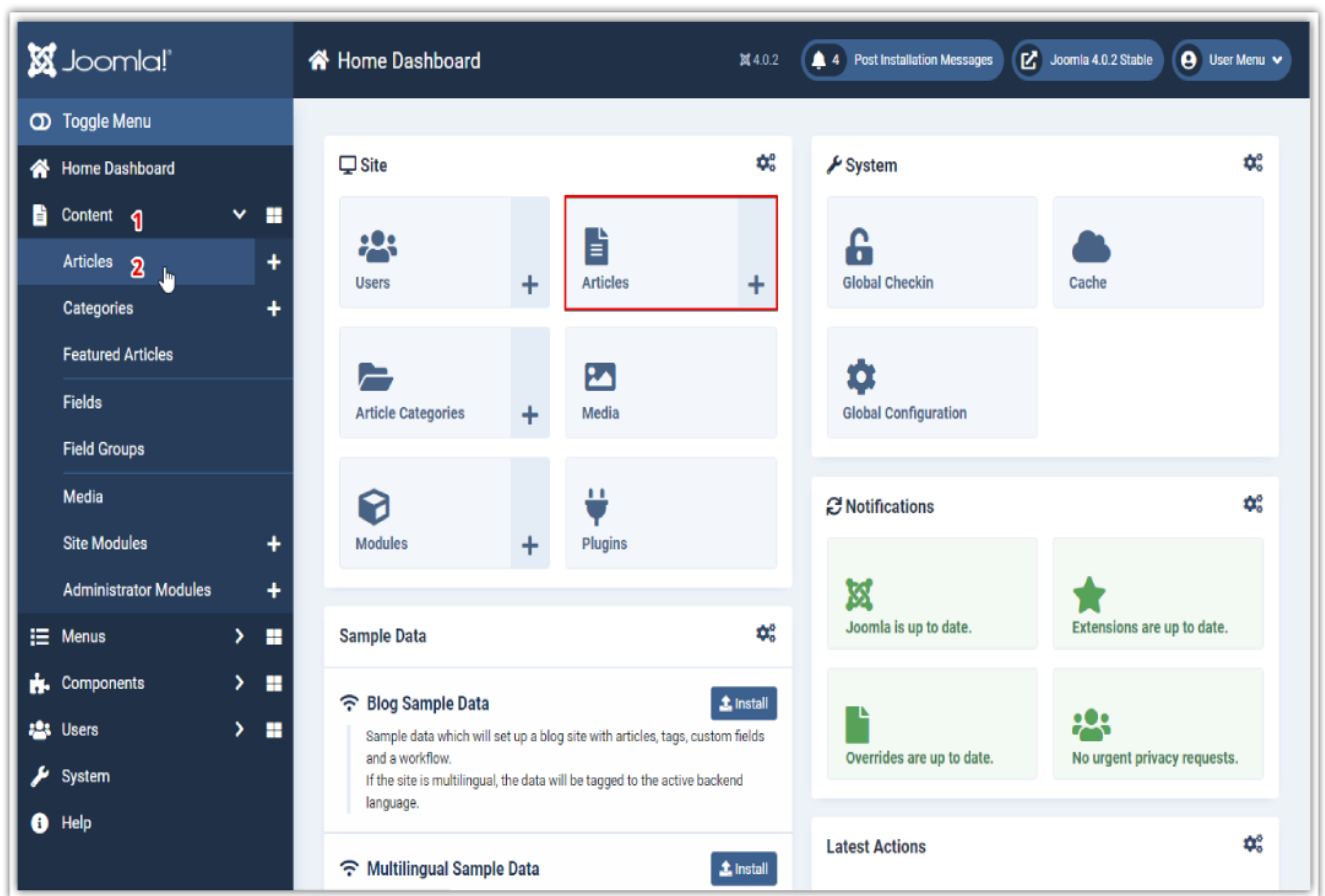


Рисунок 3.3 – Приклад інтерфейсу Joomla

- WordPress – дає можливість розробляти проекти будь-якої складності, оскільки має достатню розширюваність (рис 3.4).

Цей програмний засіб є повністю безкоштовним.

У ньому можна реалізувати ресурси широкого призначення: від звичайного блогу до інтернет-магазину [23].

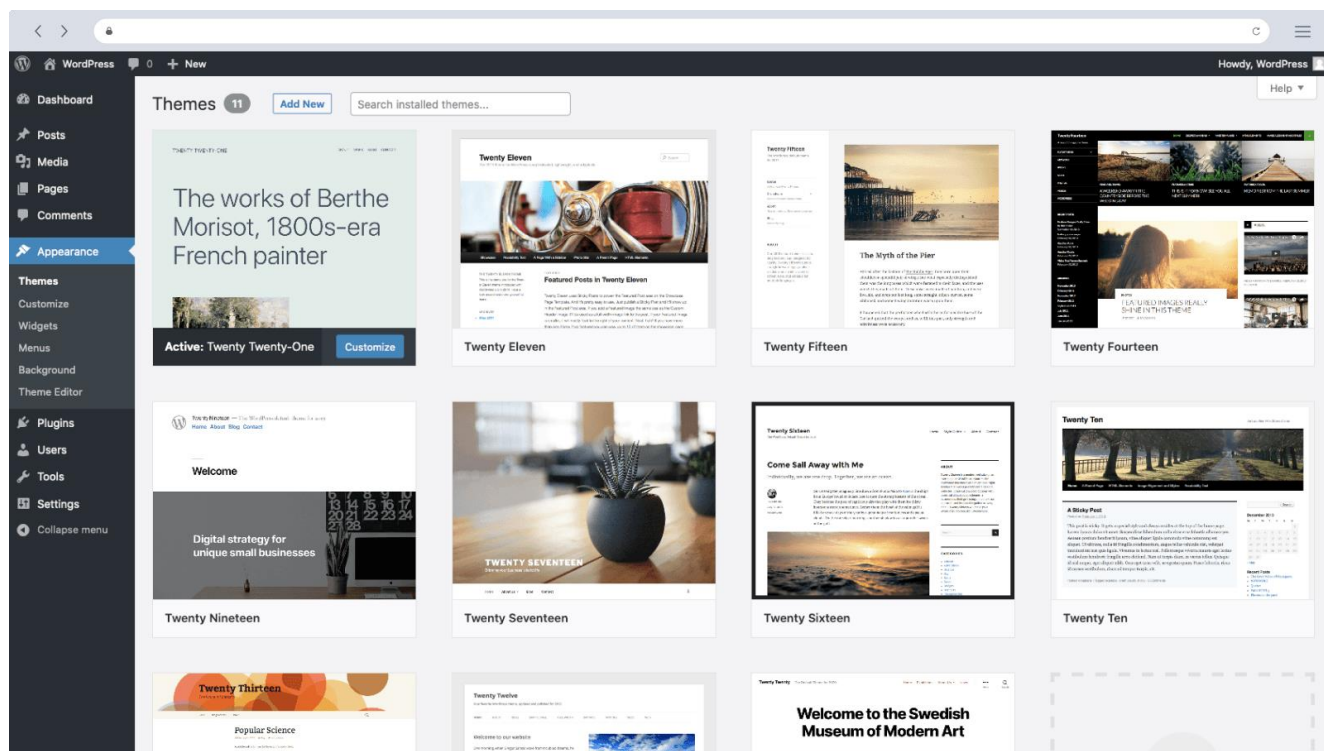


Рисунок 3.3 – Приклад інтерфейсу WordPress

Фреймворк – це структура програмного забезпечення, яка робить легшим розробку та поєднання різних складників масштабного програмного проекту. Якщо порівнювати з бібліотеками, які акумулюють набір так званих підпрограм, котрі близькі один до одного за функціональністю, то фреймворк включає в себе велику кількість різних за призначенням бібліотек.

Фреймворк робить розробку та написання коду простою, оскільки велика кількість фреймворків значно спрощує доступ до баз даних.

Для створення фронтенд частини часто використовують:

- Angular – головною метою якого є створення лише одно сторінкових сайтів. Він має високу продуктивність, а також не вимагає особливих знань у програмуванні. На просторах Інтернету є безліч інформації про цю мову програмування, але в більшості це іноземні сайти.
- Vue.js — це сучасний фреймворк, який підходить для розробки користувацьких інтересів. Якщо подивитися з іншого боку, то Vue підходить і для створення більш складних односторінкових додатків.

- React – основна його перевага – це багаторазові компоненти. Розробник створює якісь частини коду, які може поєднати між собою, щоб сформувати вже більш великі компоненти чи використати їх як окремий елемент інтерфейсу. Основний плюс використання React – це те, що створені компоненти можна використовувати повторно.

Отже, було обрано варіант розробити веб-систему за допомогою мов програмування HTML, CSS, JavaScript та за допомогою фреймворку React.

3.2 Розробка бази даних

База даних є однією з основних складових програмування сайту, адже її використовують для зберігання різної потрібної інформації. Для того, щоб розробити програмне забезпечення, потрібно добре спроектувати базу даних, бо саме тут будуть зберігатися дані, які необхідні для повноцінної роботи сайту. Є безліч систем керування базами даних.

Для розробки бази даних у веб-системі для моніторингу та підтримки навчально-виховного процесу навчального закладу було обрано нереляційну систему керування базами даних (СКБД), а саме MongoDB. Це система керування базами даних із сімейства NoSQL з відкритим кодом, в основі якої лежить документо-орієнтована модель, яка розроблена для високої продуктивності, доступності та простої масштабованості [24]. MongoDB має велику кількість переваг:

- швидкість – має високу продуктивність при реалізації легких запитів;
- легко масштабується;
- СКБД не вимагає окремого адміністратора бази даних й, оскільки вона проста у використанні, то нею можуть легко користуватися адміністратори та розробники;
- кросплатформеність – оскільки вона написана на C++, то її просто перенести на будь-які різноманітні платформи [25].

Для моделювання об'єктів була використана бібліотека Mongoose. Це бібліотека JavaScript, яка часто використовується в додатку Node.js з базою даних MongoDB. На рисунку 3.4 наведено код для налаштування таблиць бази даних.

```
1  const {Schema, model, Types} = require('mongoose')
2
3  const schema = new Schema({
4    email: {type: String, required: true, unique: true},
5    password: {type: String, required: true},
6    user_name: {type: String, required: true},
7    user_id: {type: Number}
8  })
9  module.exports = model('User', schema);
10
11 const {Schema, model, Types} = require('mongoose')
12 const schema = new Schema({
13   number: {type: String, required: true},
14   name: {type: String, required: true},
15   sex: {type: String, required: true},
16   birth_date: {type: String, required: true},
17   parents: [{type: String}],
18   phone: {type: String, required: true},
19   address: {type: String, required: true},
20   teacher_id: {type: Number, required: true}
21 })
22 module.exports = model('Statement', schema);
23
24 const {Schema, model, Types} = require('mongoose')
25 const schema = new Schema({
26   classroom: {type: Number, required: true},
27   teacher_id: {type: Number, required: true},
28   teacher_name: {type: String, required: true},
29   students: [{
30     name: String,
31     stud_id: Number,
32     marks: [{
33       sub_id: Number,
34       mark: String
35     }],
36   }],
37   subjects: [{
38     name: String,
39     sub_id: Number
40   }],
41   class_id: {type: Number, required: true},
42 })
43 module.exports = model('Progress', schema);
```

length: 135

Рисунок 3.4 – Лістинг бази даних

3.3 Розробка модуля електронного журналу

Основною задачею веб-системи є моніторинг та підтримка навчально-виховного процесу навчального закладу, тому ключовою тут є розробка електронного журналу. Щоб відслідковувати успішність учнів у різних класах, було розроблено різні сторінки журналів. Оскільки завантаження повного журналу з усіма даними може негативно вплинути на час очікування, то було вирішено розбити журнал на класи, до яких має доступ відповідно директор, адміністрація закладу і вчителі, які можуть заходити лише під своїм логіном і паролем. Нижче наведено код електронного журналу в системі (рис. 3.5).

```
1  const {Router} = require('express')
2  const Statement = require('../models/Statement')
3  const Progress = require('../models/Progress')
4  const router = Router()
5
6  router.get(
7    '/statement/:id',
8    async (req, res) => {
9      try {
10         const statementId = req.params.id;
11
12         const statementData = await Statement.find({ teacher_id: statementId});
13
14         if (!statementData) {
15             return res.status(400).json({ message: 'not found'})
16         }
17
18         res.json({data: statementData})
19     } catch (e) {
20         console.log(e);
21         res.status(500).json({ message: 'Something went wrong, try again' })
22     }
23 }
24 })
25 router.get(
26   '/progress/:id',
27   async (req, res) => {
28     try {
29       const progressId = req.params.id;
30
31       const progressData = await Progress.findOne({ teacher_id: progressId});
32
33       if (!progressData) {
34           return res.status(400).json({ message: 'not found'})
35       }
36       res.json({data: progressData})
37     } catch (e) {
38         console.log(e);
39         res.status(500).json({ message: 'Something went wrong, try again' })
40     }
41 }
42 })
43 module.exports = router
```

Рисунок 3.5 – Лістинг модуля електронного журналу

3.4 Розробка клієнтської частини

Під час розробки веб-системи було розроблено клієнтську частину. Отриманий електронний журнал передається на клієнтську сторону через API. Клієнтська частина фільтрується за класами і учнями, що дає змогу показати повний функціонал електронного журналу.

Для розробки клієнтської частини був задіяний компонентний підхід розробки. Компонентний підхід – це поєднання програмних компонентів та опис їх взаємодії між собою. Клієнтська частина написана за допомогою бібліотеки JavaScript – React, яка слугує для створення інтерфейсів користувача . Приклад коду наведено нижче (рис 3.6 та рис. 3.7).

```

1  import React, {useState, useEffect} from 'react';
2  import StatementTable from '../components/statement-table';
3  import ProgressTable from '../components/progress-table';
4
5  const ProgressPage = () => {
6
7      const userId = JSON.parse(localStorage.getItem('adminData')).userId;
8
9      const [statementData, setStatementData] = useState(null);
10     const [progressData, setProgressData] = useState(null);
11
12
13
14     useEffect(() => {
15         fetch(/api/school/statement/${userId})
16             .then(res => res.json())
17             .then(json => {
18                 setStatementData(json.data)
19             })
20
21         fetch(/api/school/progress/${userId})
22             .then(res => res.json())
23             .then(json => {
24                 setProgressData(json.data)
25             })
26     }, [])
27
28
29
30     if (!progressData || !statementData) {
31         return(
32             <p>
33                 loading
34             </p>
35         )
36     }
37
38     const st = statementData.length ? <StatementTable data={statementData} titles={["№", "Homep",
39
40     return (
41         <div className='my_table'>
42             {st}

```

Рисунок 3.6 – Лістинг клієнтської частини

```

43     <h3>Клас {progressData.classroom}</h3>
44     <h5>{progressData.teacher_name}</h5>
45     <ProgressTable data={progressData}/>
46   </div>
47   );
48 }
49
50 export default ProgressPage
51
52 import React from 'react';
53
54 const ProgressTable = ({data}) => {
55
56   const tableHeaders = data.subjects.map((el, idx) => {
57     return (
58       <th key={`${el}-${idx}`}>
59         {el.name}
60       </th>
61     )
62   })
63
64   const tableBody = data.students.map((el, idx) => {
65     const bodyData = []
66     bodyData.push(
67       <td key={`-${idx}`}>{el.name}</td>
68     );
69
70     data.subjects.map((s, i) => {
71       for (const mark of el.marks) {
72         if (s.sub_id === mark.sub_id) {
73           bodyData.push(
74             <td key={`${mark.mark}-${i}`}>{mark.mark}</td>
75           );
76         }
77       }
78     })
79
80     return (
81       <tr key={`${el}-${idx}`}>
82         <td>{idx+1}</td>
83         {bodyData}
84       </tr>

```

Рисунок 3.7 – Продовження лістингу клієнтської частини

3.5 Розробка головного файлу сервера

Під головним файлом сервера, мається на увазі керування усіма модулями та файлом конфігурацій. Файл сервера виконується через командну стрічку за допомогою визначених команд. Після виконання деяких операцій сервер завантажується. Приклад коду наведено на рис. 3.8.

```
1  const express = require('express')
2  const config = require('config')
3  const mongoose = require('mongoose')
4
5  const app = express()
6
7  app.use(express.json({ extended: true }))
8
9  app.use('/api/auth', require('./routes/auth'))
10 app.use('/api/school', require('./routes/school'))
11
12 const PORT = config.get('port') || 5000
13
14 async function start() {
15   try {
16     await mongoose.connect(config.get('mongoUrl'), {
17       ...
18     })
19   }
20   app.listen(PORT, () => {
21     console.log(`App has been started on port ${PORT}...`);
22   })
23 } catch (e) {
24   console.log('Server Error', e.message)
25   process.exit(1)
26 }
27 }
28
29 start()
```

Рисунок 3.8 – Лістинг файлу сервера

3.6 Висновки

У третьому розділі було описано вибір мови програмування та технологій для розробки програмного продукту. Проведено аналіз потреб програмного додатку та обрано мову програмування JavaScript.

Для розробки бази даних було обрано MongoDB.

Також було розроблено модуль електронного журналу. Для формування структури сайту було обрано HTML та CSS.

4 ТЕСТУВАННЯ ВЕБ-СИСТЕМИ

4.1 Тестування веб-системи

Створення сайту – це процес, що поєднує етапи розробки від розробки структури, моделі, алгоритмів роботи до програмної реалізації. Остання стадія усього процесу – веб-тестування сайтів. Тестування сайту відіграє важливу роль в створенні ресурсу, оскільки від якості роботи залежить життєвий цикл проєкту. Практика доводить, що більшість розробників не надають належну увагу цьому етапу. Це призводить до появи помилок, які обумовлюють колосальні витрати часу, сил і грошей. Ресурс, який працює некоректно, викликає у користувачів негатив, в результаті довіра до компанії падає, що негативно впливає на її репутацію.

UI testing – це засіб тестування графічного інтерфейсу користувача, що проводить перевірку веб-сайту на відповідні вимоги. Сайт перевіряється за усіма вимогами до графічного інтерфейсу з огляду на зрозумілість користувачу, на професійний вигляд, на зручність виклику тієї чи іншої сторінки сайту. Здійснюється перевірка вибору кольорової гами щодо її комфортності для очей та виклику позитивних емоцій у користувача. Таке тестування допомагає зекономити час і гроші, а також збільшує надійність програмного продукту [26].

Основні пункти тестування (UI):

- тестування на відповідність стандартам графічних інтерфейсів;
- тестування коректності написання тексту і чи текст є відформатованим;
- тестування сумісності з усіма Інтернет браузерами та їх версіями;
- тестування функціональності частини інтерфейсу.

Метою тестування є перевірка ресурсу на відповідність та функціональність за відповідних вимог. Зазвичай тестування проводиться до того часу, поки сайт не буде відповідати вимогам замовника.

Початкова сторінка веб-ресурсу та її дизайн має одразу дати зрозуміти, куди користувач потрапив і що він може отримати в результаті відвідування даного сайту. Будь-яка не потрібна інформація буде відволікати відвідувача та заважатиме зосередити увагу на тому, що йому потрібно.

Головна сторінка веб-системи навчально-виховного комплексу «Малинківський заклад освіти І ст.-сад», на якій зображено девіз, оголошення, новини та фото школи (рис. 4.1).

Також на кожній із сторінок зберігається шапка сайту, на якій є фото території школи.

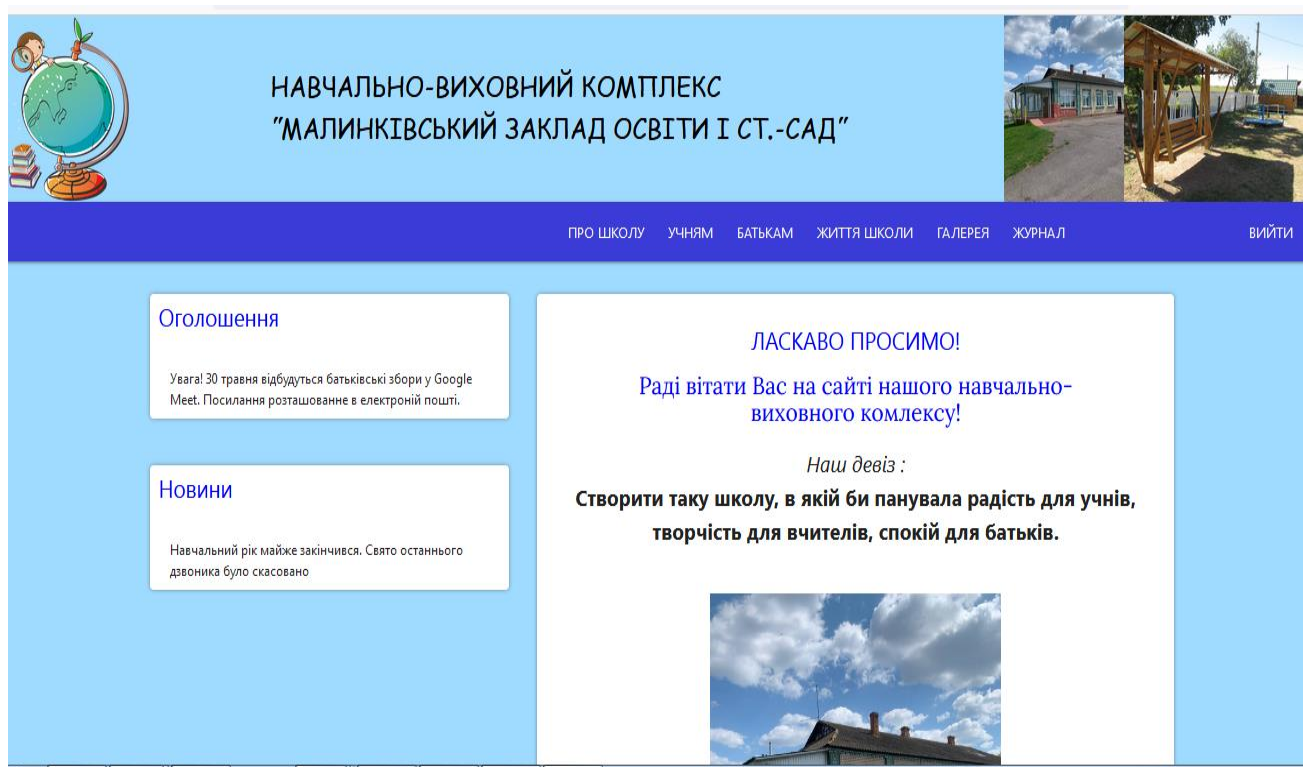


Рисунок 4.1 – Головна сторінка веб-системи навчально-виховного комплексу «Малинківський заклад освіти І ст. – сад»

Сторінка «Про школу» надає відповідну інформацію про навчально-виховний комплекс та відкриває навігацію і перехід до сторінок, які мають більшу

інформативність і спеціалізовану орієнтованість інформаційного забезпечення (рис.4.2).

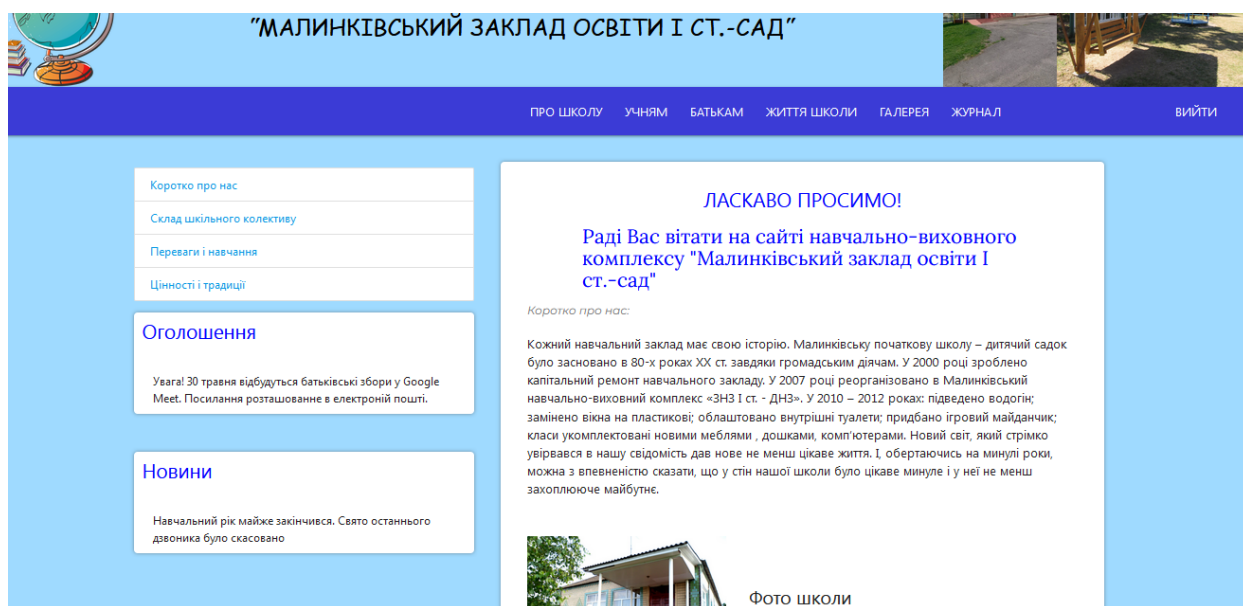


Рисунок 4.2 – Веб-сторінка «Про школу»

Також на кожній з обраних сторінок є юридична адреса та контактний номер, за допомогою якого можна зв'язатися з адміністрацією навчально-виховного комплексу (рис.4.3).

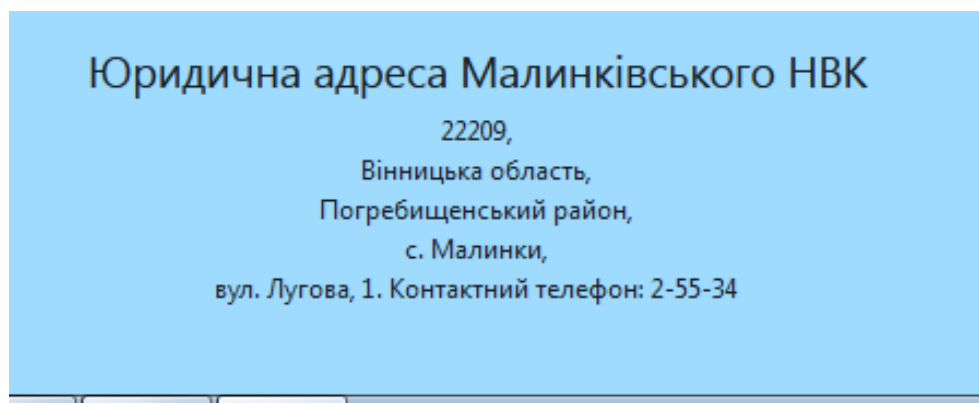


Рисунок 4.3 – Юридична адреса навчально-виховного комплексу

Веб-сторінка «Учням» дає доступ до цікавих матеріалів, які потрібні учням початкових класів (рис. 4.4).

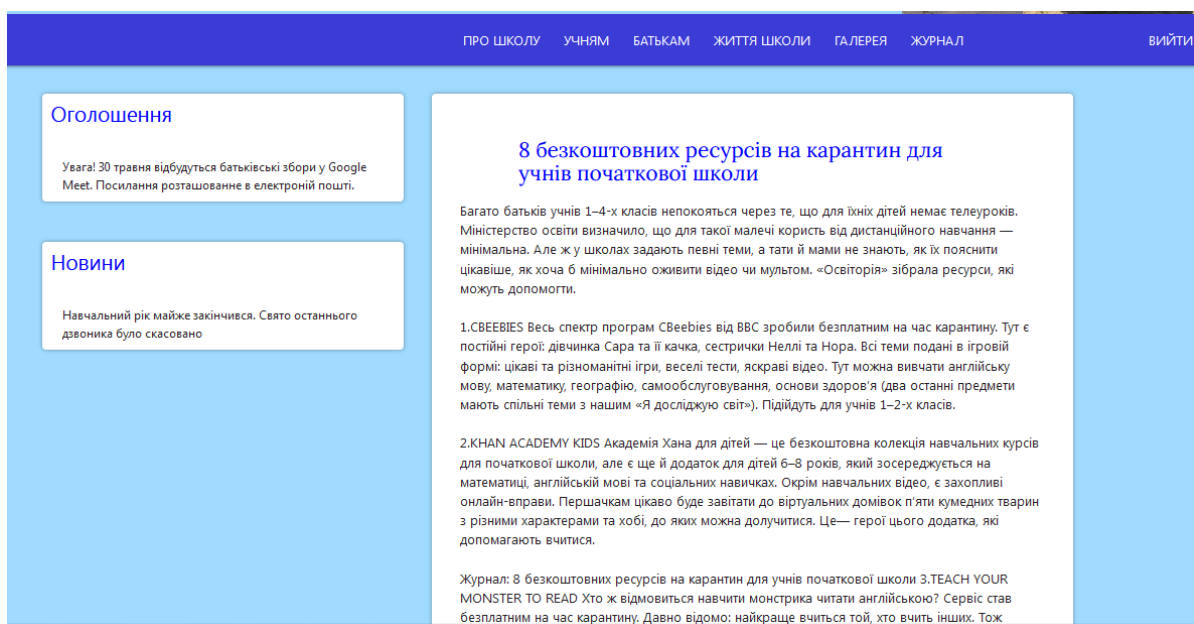


Рисунок 4.4 – Веб-сторінка «Учням»

На сторінці «Батькам» презентується актуальна інформація, а також поради щодо виховання дитини (рис. 4.5).

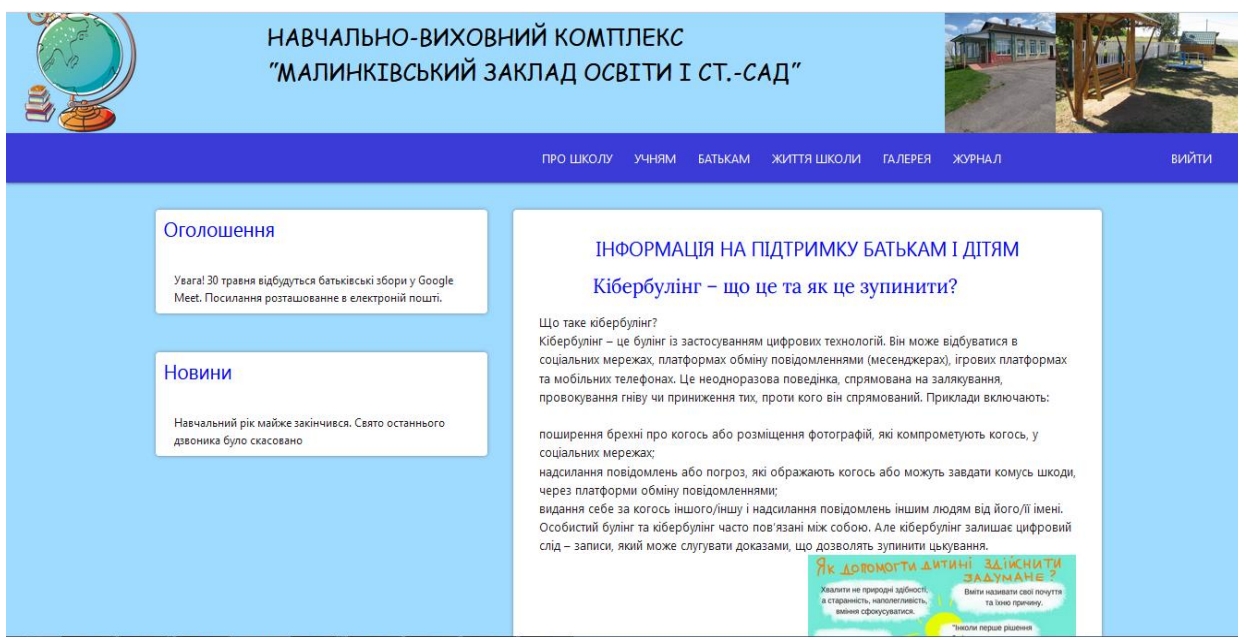


Рисунок 4.5 – Веб-сторінка «Батькам»

Наступна сторінка веб-системи – це «Галерея», на якій висвітлені різні фото з проведених заходів та свят (рис. 4.6).

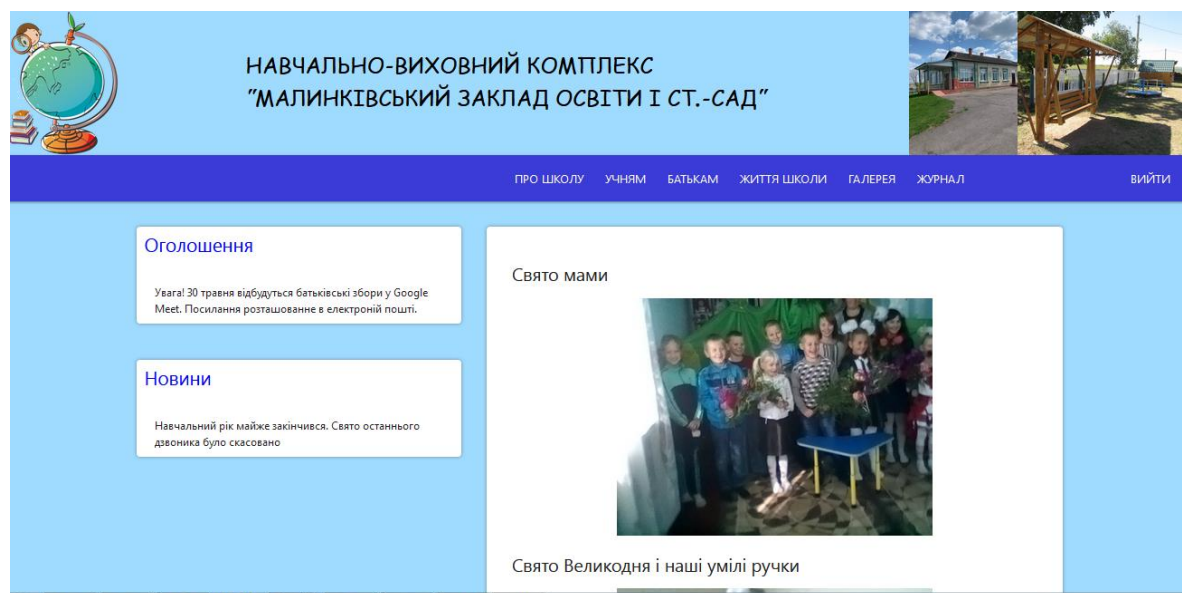


Рисунок 4.6 – Веб-сторінка «Галерея»

Сторінка «Життя школи» надає можливість відвідувачам сайту побачити всі заходи, свята, які проводяться чи заплановані для проведення у навчально-виховному комплексі. На цій сторінці розташовані не тільки фото, але є текст, який легко читається та є зрозумілим (рис. 4.7).

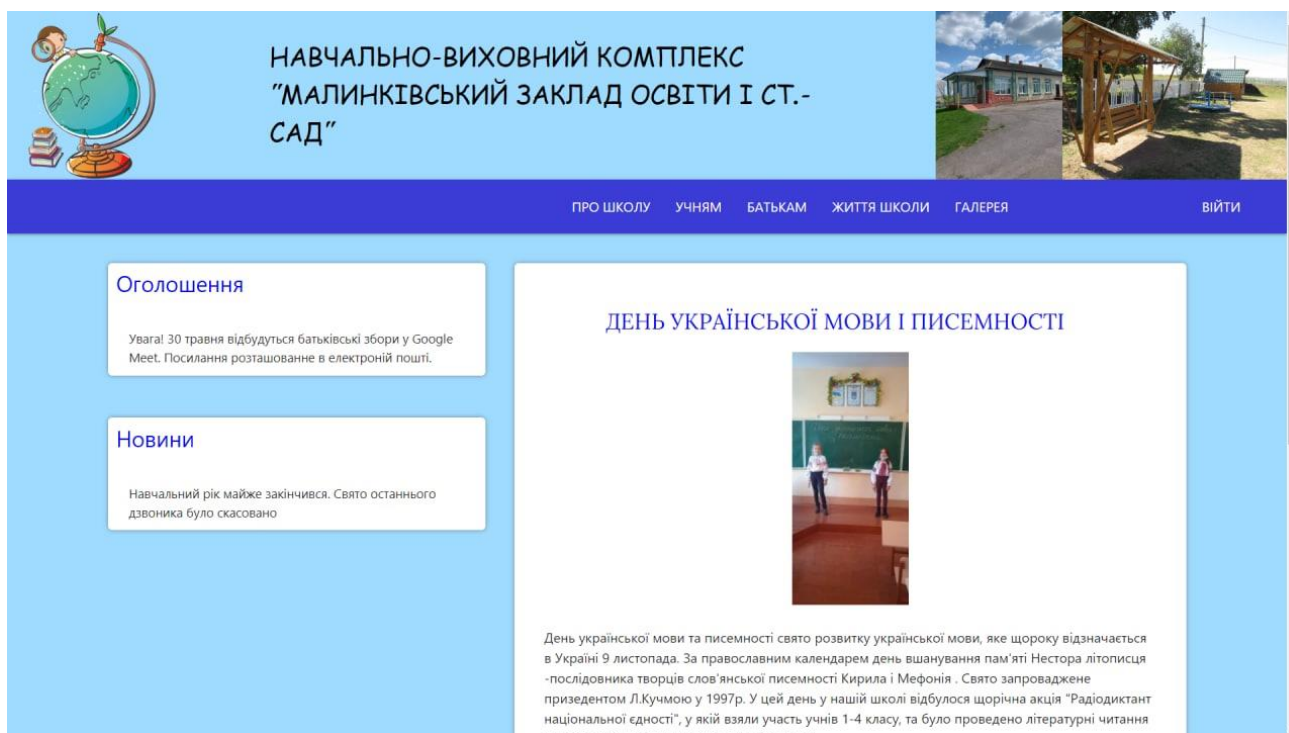


Рисунок 4.7 – Веб-сторінка «Життя школи»

Також на сайті розроблена зручна навігація, при якій з легкістю можна переходити зі сторінки на сторінку (рис.4.8).

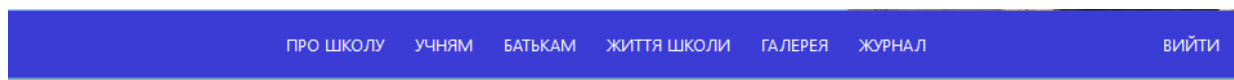


Рисунок 4.8 – Навігація сайту

Веб-сторінка «Журнал» при вході на сайт не доступна усім користувачам. Вона доступна лише авторизованим особам які мають логін і пароль. Щоб цей пункт меню з'явився, потрібно зареєструватися й авторизуватися. Вікно авторизації – легке і зрозуміле (рис 4.9). Після введення логіну і паролю (рис. 4.10) отримуємо доступ до журналу.

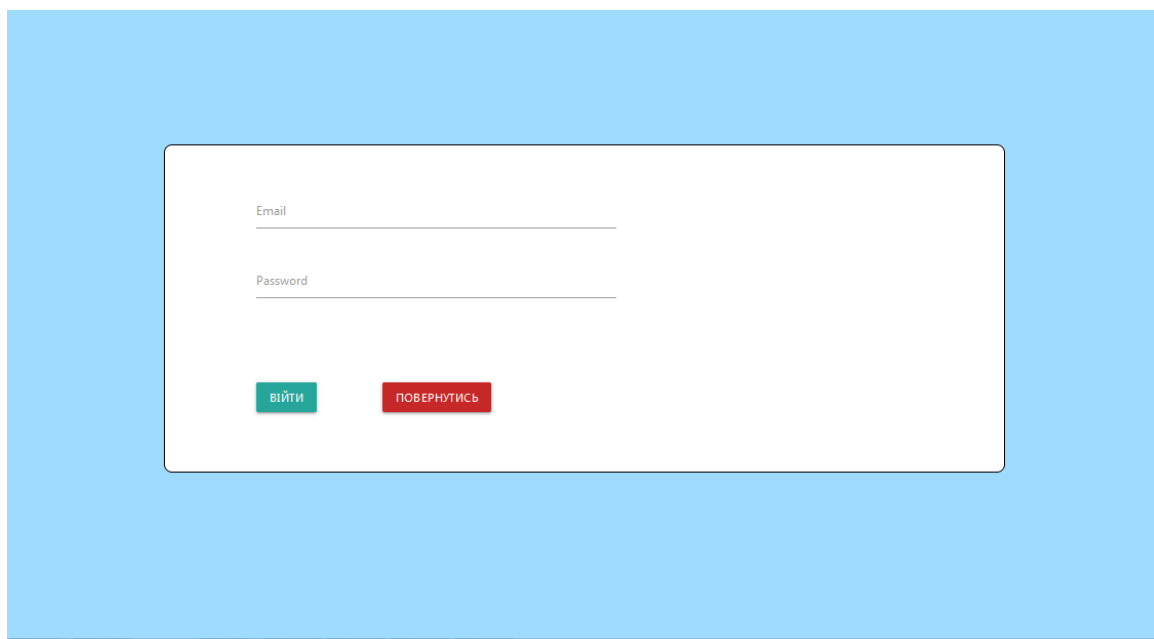


Рисунок 4.9 – Вікно авторизації

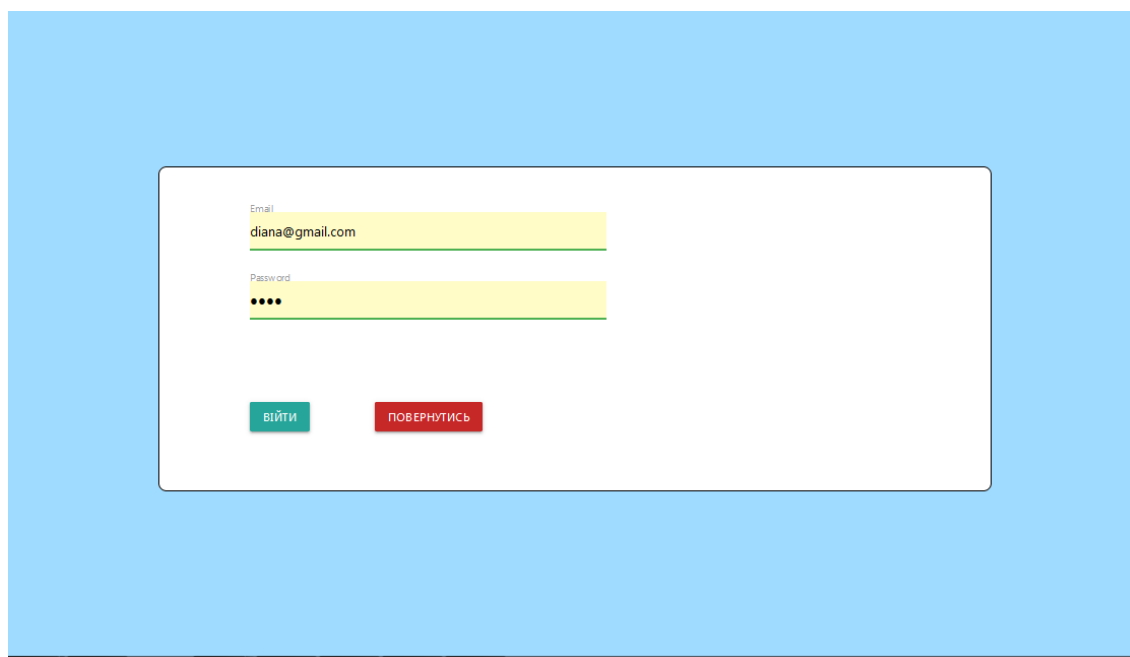


Рисунок 4.10 – Вікно авторизації із введеними даними

Якщо дані введені некоректно, веб-система повідомляє про це користувача і з'являється вікно із попередженням (рис. 4.11).

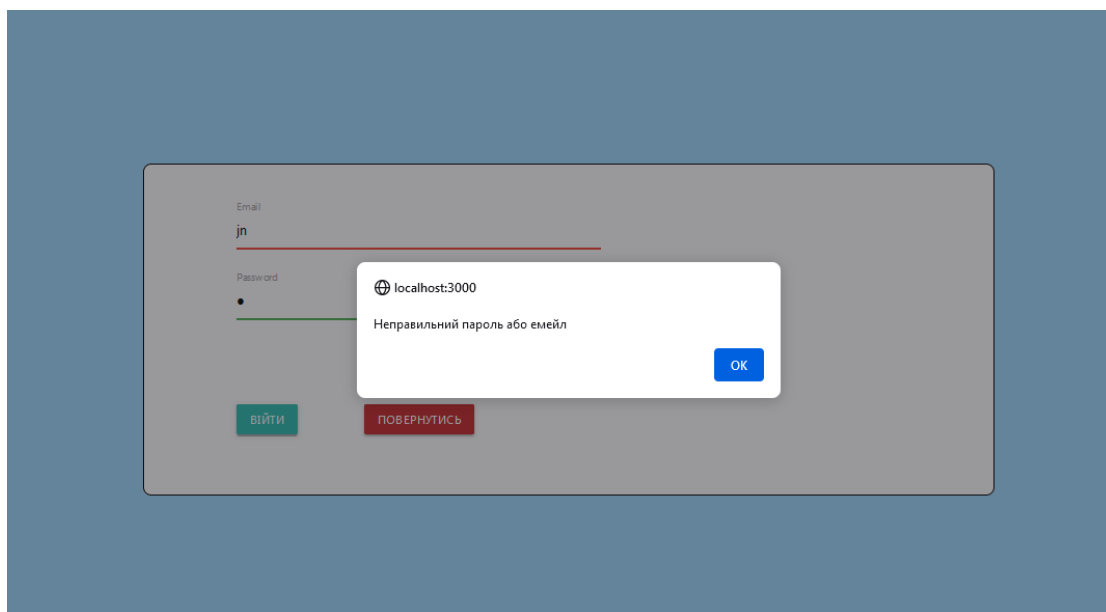


Рисунок 4.11 – Вікно авторизації із некоректними даними

При авторизації з'являється сторінка «Журнал», при переході на яку адміністрація закладу і вчителі отримують доступ до електронного журналу, списку учнів та їх даних (рис 4.12).

№	Номер	ПІБ	Стать	Дата народження	ПІБ батьків	Телефон	Адреса
1	G4	Гнатенко Антон Олександрович	чол	03.09.2011	Гнатенко Олександр Миколайович Гнатенко Ольга	0963433161	с. Малинки вул. Нижньо-Садова Вінницька обл.
2	G5	Гнатиш Анастасія Анатоліївна	жін	02.10.2012	Гнатиш Анатолій Анатолійович Гнатиш Олена Іванівна	0986535890	с. Малинки вул. Нижньо-Садова Вінницька обл.
3	K5	Кравченко Богдана Іванівна	жін	29.11.2011	Кравченко Іван Іванович Кравченко Альона Олександрівна		с. Малинки вул. Нижньо-Садова Вінницька обл.
4	M12	Мінич Ілля Борисович	чол	18.01.2012	Мінич Борис Васильович Мінич Анжела Василенна	0979041981	с. Малинки вул. Нижньо-Садова Вінницька обл.
5	Ш2	Школьна Ульяна Сергіївна	жін	30.01.2012	Школьний Сергій Борисович Хоменко Валентина Іванівна	0989174492	с. Малинки вул. Нижньо-Садова Вінницька обл.

Рисунок 4.12 – Дані про учнів та батьків

Батьки можуть переглядати результати поточної успішності своїх дітей. Вони не мають прав доступу до редагування інформаційного наповнення сайту.

Учитель може вносити корективи оцінок і предметів та заповнювати інформаційну базу свого класу (рис.4.13).

Клас 1
Турчанівська Надія Петрівна

№ Учні	Українська мова і літературне читання	Англійська мова	Математика	Я досліджую світ (інтегрований курс)	Дизайн і технології	Мистецтво	Фізична культура
1 Гнатенко Антон Олексійович	добре	відмінно	відмінно	добре	відмінно	добре	відмінно
2 Гнатиш Анастасія Анатоліївна	добре	добре	добре	добре	добре	добре	добре
3 Кравченко Богдана Іванівна	відмінно	відмінно	добре	добре	відмінно	відмінно	відмінно
4 Мінич Ілля Борисович	відмінно	відмінно	відмінно	відмінно	відмінно	відмінно	відмінно
5 Школьна Уляна Сергіївна	відмінно	добре	добре	відмінно	добре	відмінно	добре

Юридична адреса Малинківського НВК
22209,
Вінницька область,
Погребищенський район,
с. Малинки,
вул. Лугова, 1. Контактний телефон: 2-55-34

Рисунок 4.13 – Електронний журнал 1 класу

**НАВЧАЛЬНО-ВИХОВНИЙ КОМПЛЕКС
"МАЛИНКІВСЬКИЙ ЗАКЛАД ОСВІТИ І СТ.-САД"**

ПРО ШКОЛУ УЧНЯМ БАТЬКАМ ЖИТТЯ ШКОЛИ ГАЛЕРЕЯ ЖУРНАЛ ВИЙТИ

Клас 2
Порошенко Ніна Сергіївна

№ Учні	Українська мова і літературне читання	Англійська мова	Математика	Я досліджую світ (інтегрований курс)	Музичне мистецтво	Образотворче мистецтво	Фізкультура
1 Тодошак Анна Василівна	відмінно	добре	добре	добре	відмінно	добре	відмінно
2 Кушнір Олександр Сергійович	добре	добре	добре	добре	відмінно	відмінно	відмінно
3 Кушнір Володимир Сергійович	добре	добре	відмінно	добре	відмінно	відмінно	відмінно
4 Польгуль Богдан Ігорович	добре	відмінно	відмінно	відмінно	відмінно	добре	добре
5 Маринич Анна Петрівна	відмінно	добре	відмінно	відмінно	добре	добре	добре

Рисунок 4.14 – Електронний журнал 2 класу

Ще на головній та інших сторінках є «Оголошення» і «Новини», де розміщують актуальну інформацію для батьків та вихованців навчально-виховного комплексу (рис. 4.15).

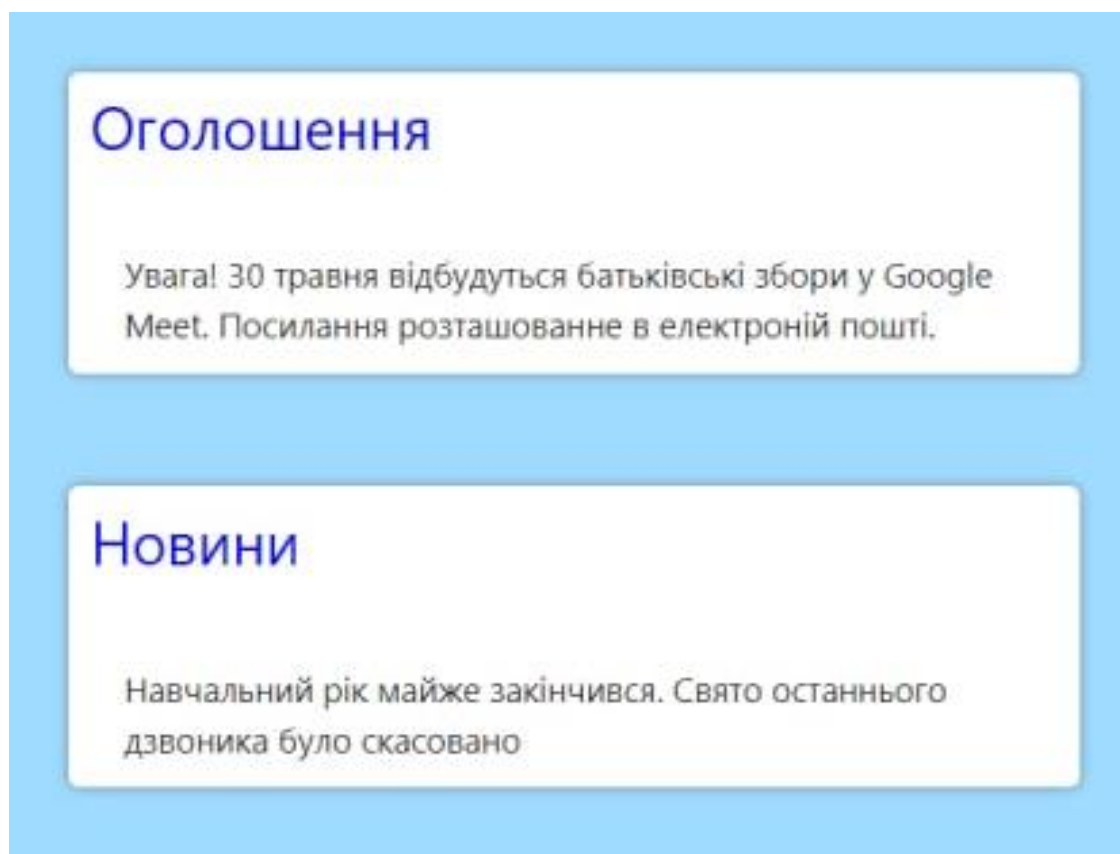


Рисунок 4.15 – «Оголошення» та «Новини»

Сайт навчально-виховного комплексу «Малинківський заклад освіти І ст. – сад» був протестований на різних браузерях, таких як Google Chrome та Mozilla Firefox.

Результати коректної роботи веб-ресурсу в різних браузерях підтверджують працездатність розробки (рис 4.16 та рис. 4.17).

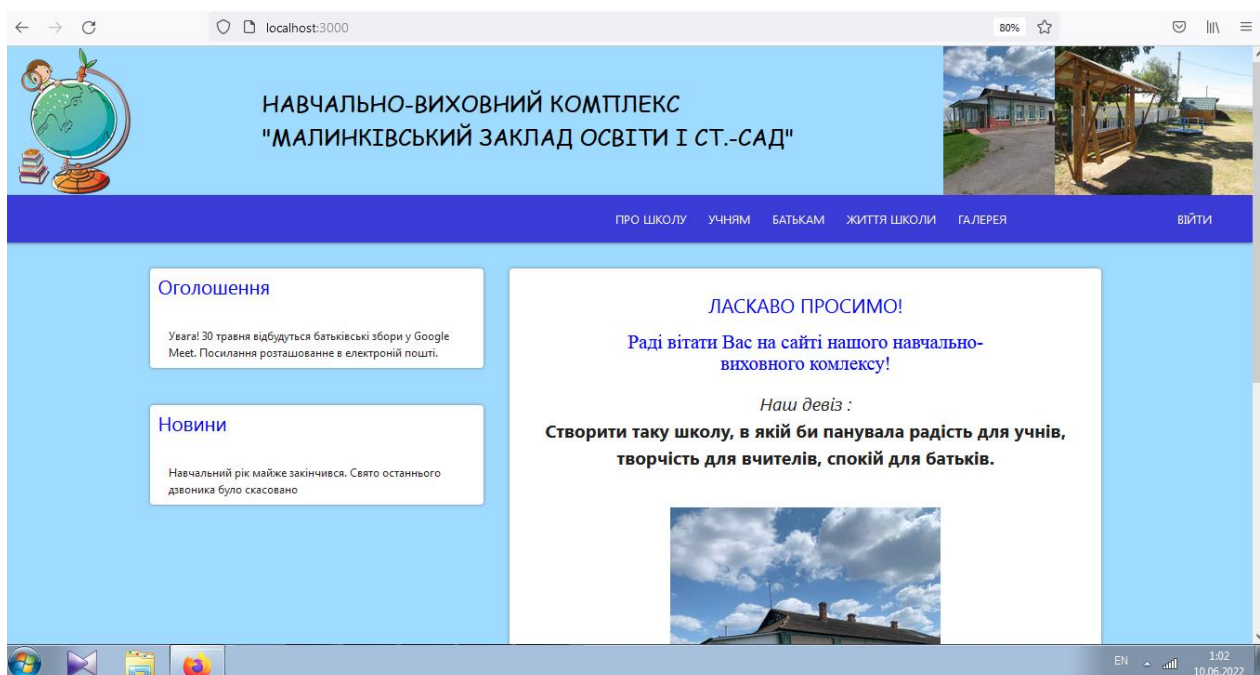


Рисунок 4.16 – Тестування в браузері Mozilla Firefox

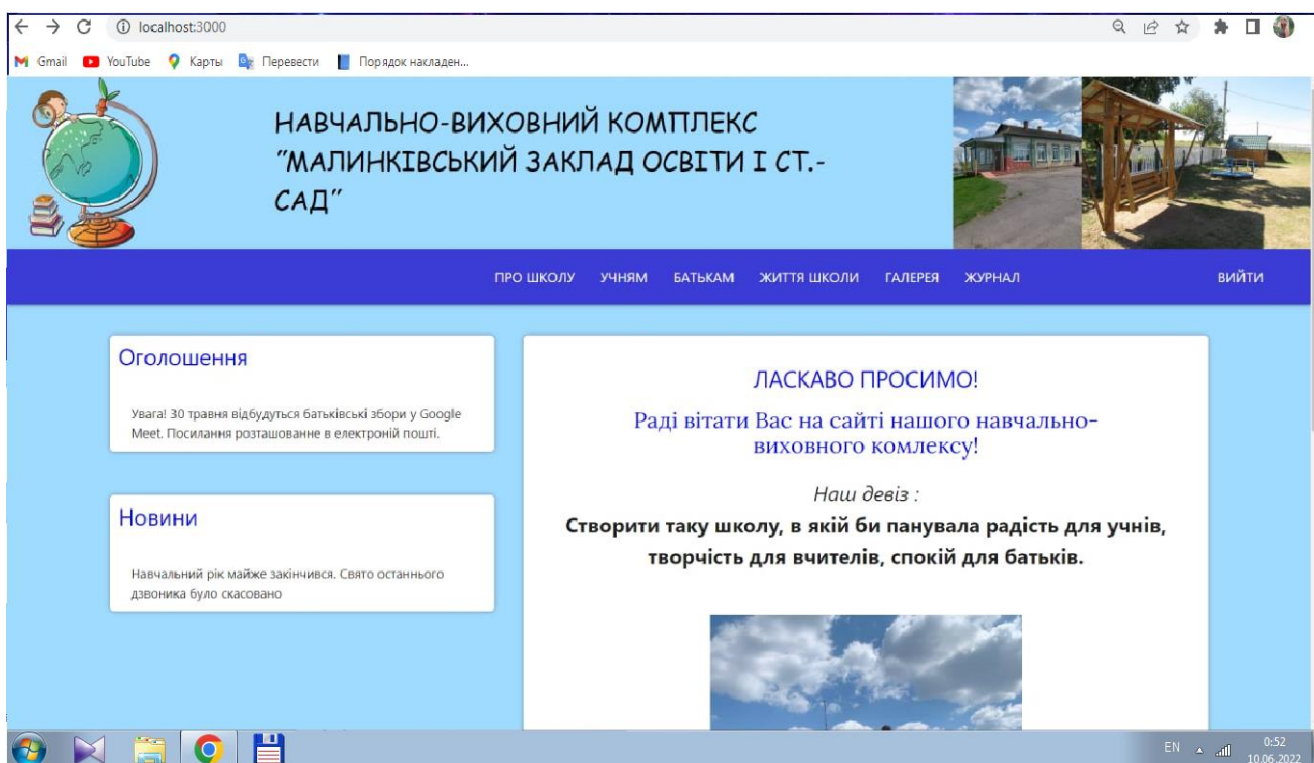


Рисунок 4.17 – Тестування в браузері Google Chrome

У наш час сучасні технології розвиваються швидко, а сучасний користувач Інтернету є досить обізнаним з роботою в Інтернеті. Користувач, як правило, є

досить вибагливим, не любить чекати, він любить комфорт, тому дизайн сайту є досить важливим з точки зору його розвитку і просування. Якщо відвідувачу щось не сподобається на сайті, він відразу його покине, тому UI-тестування є обов'язково складовою частиною розробки.

4.2 Розробка інструкції користувача

Одним із важливих видів тестування є функціональне тестування. Функціональне тестування сайту – це вид перевірки, що показує неправильну роботу функціоналу програми.

Також варто здійснювати тестування продуктивності, яке, в основному, є тестуванням навантаження на сервер. Таке тестування дозволяє визначити оперативність і стабільність роботи веб-системи, адже веб-ресурс повинен витримувати високі навантаження. Тестування продуктивності веб-системи перевіряється в більшості випадків автоматично, використовуючи спеціальні програми. Дане тестування дає можливість перевірити, наскільки ресурс буде коректно працювати при великих навантаженнях.

Тестування інтерфейсу – це тестування графічного інтерфейсу користувача, яке передбачає перевірку сайту на відповідність вимогам до графічного інтерфейсу, наскільки професійно він виглядає, чи виконаний дизайн в єдиному стилі [27].

Інструкція для відвідувача веб-системи окреслює низку технічних вимог для роботи програмного продукту. Подробиці щодо мінімальної та рекомендованої конфігурації серверного комп'ютера зведено в таблиці 4.1 та 4.2.

Таблиця 4.1 – Мінімальна конфігурація:

Тип процесора	32-розрядний (x86) або 64-розрядний (x64) процесор з тактовою частотою 1 ГГц
Об'єм оперативної пам'яті	2 ГБ для 32-розрядної системи

Продовження таблиці 4.1

Місце на жорсткому диску	Від 1 ГБ
Операційна система	Windows 7-10

Таблиця 4.2 – Рекомендована конфігурація:

Тип процесора	32-розрядний (x86) або <u>64-розрядний (x64)</u> процесор з тактовою частотою 2 ГГц
Об'єм оперативної пам'яті	8 ГБ для 32-розрядної системи
Розмір жорсткого диску	10 ГБ
Операційна система	Windows 7-10

Для запуску програмного продукту потрібно завантажити Node.js і завантажити саму веб-систему. Наступним кроком потрібно відкрити термінал і запусити сервер.

Після пуску системи користувач може авторизуватися в системі та перейти на потрібну вкладку.

4.3 Висновки

У четвертому розділі було розглянуто методи тестування та проведено основні етапи тестування веб-системи моніторингу та підтримки навчально-виховного процесу закладу освіти. Проведено тестування форми авторизації. Веб-сайт є зручним для користувача, має зручну навігаційну панель та зрозумілу логічну структуру. Також було розроблено інструкцію користувача щодо встановлення та початкової експлуатації програмного продукту.

ВИСНОВКИ

У бакалаврській дипломній роботі було розроблено веб-систему для моніторингу та підтримки навчально-виховного комплексу. Для розробки було використано середовище програмної розробки Microsoft Visual Code.

Під час виконання бакалаврської дипломної роботи було проаналізовано стан задачі, розглянуто основні аналоги програмного продукту, визначено їхні переваги і недоліки та порівняно з власним програмним продуктом. Опираючись на результати порівняння, було сформульовано основні задачі бакалаврської дипломної роботи.

Під час аналізу технологій розробки було обґрунтовано вибір мови програмування JavaScript. Серверна частина розроблена за допомогою Node.js. А для зберігання даних використано базу даних MongoDB.

У бакалаврській дипломній роботі було зроблено:

- розроблено метод моніторингу та підтримки навчально-виховного процесу у закладах освіти з використанням спеціалізованої веб-системи;
- розроблено модель роботи веб-системи;
- проаналізовано інформаційне забезпечення сайту;
- розроблено програмні засоби реалізації веб-системи моніторингу та підтримки навчально-виховного процесу;
- проведено тестування роботи веб-системи.

Тестування програми показало цілковиту працездатність програмного продукту та відповідність поставленому технічному завданню. Також було розроблено інструкцію користувача.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Веб-безпека. Різновиди веб-систем та їх основні вразливості [Електронний ресурс]. Режим доступу до ресурсу: <http://websecurity.com.ua/security/chapter1/>
2. Що таке сайт? [Електронний ресурс] – Режим доступу до ресурсу: <https://2ip.ua/ua/blog/website>
3. Войтко В.В. Особливості розробки веб-сайту навчально-виховного комплексу "Малинківський заклад освіти І ст.- САД " / В.В. Войтко, Л.М.Круподьорова, А. В. Денисюк, О. В. Гаврилюк, Н. Є. Барчук, Д. С. Лаба // Матеріали ІІ Науково-технічна конференція факультету інформаційних технологій та комп'ютерної інженерії (2022) ВНТУ, Факультет інформаційних технологій та комп'ютерної інженерії, Секція програмного забезпечення. [Електронний ресурс] – Режим доступу до ресурсу: <https://conferences.vntu.edu.ua/index.php/all-fitki/all-fitki-2022/paper/view/15097/12733>
4. Web Development Recipes: навч. посіб. / Brian P. Hogan, Chris Warren, Pragmatic Bookshelf 2 edition, 2015. – 360 ст.
5. What is a Website ? [Електронний ресурс] – Режим доступу до ресурсу: <https://www.geeksforgeeks.org/what-is-a-website/>
6. Веб-сайт комунального закладу "Заклад загальної середньої освіти І-ІІІ ступенів с. Зозулинці" [Електронний ресурс] – Режим доступу до ресурсу: <http://zsl.khedu.vn.ua/>
7. Веб-сайт Лукашівського навчально - виховного комплексу [Електронний ресурс] – Режим доступу до ресурсу: <https://lukashvskij-nvk.mozello.com/>
8. Веб-сайт Лавриківського закладу загальної середньої освіти І-ІІІ ступенів комплексу [Електронний ресурс] – Режим доступу до ресурсу: <https://sites.google.com/site/lavrykivschool/golovna>

9. Веб-сайт ЗЗСО I-III ступенів ім. Г.Т. Танцюри с. Зятківці [Електронний ресурс] – Режим доступу до ресурсу: <http://ztk.gnedu.vn.ua/>
10. Курс лекцій з дисципліни «Веб-проекування»/ Уклад.: О.П. Цурін. – К.: НТУУ «КПІ», 2011 р. ст.7-19, 34-47
11. Джок Бехт, Рентабельний Web-дизайн. – К., 2003 – 236 с
12. "Static vs Dynamic Websites: Pros and Cons" [Електронний ресурс] – Режим доступу до ресурсу: <https://wsvincent.com/static-vs-dynamic-websites-pros-and-cons>
13. Kronin D. Kuper A. Reyman R. Alan Cooper on the interface. Fundamentals of interaction / Alan Kuper ob interfeyse. Osnovy proektirovaniya vzaimodeystviya Paperback – January 1, 2009
14. Юзабіліті веб-сайту. Основні принципи юзабіліті: від особливостей дизайну до контенту, від інтерфейсу форми замовлення до пошуку. Можливості тестування. [Електронний ресурс] – Режим доступу: <https://www.ashmanov.com/education/articles/yuzabiliti/>
15. Мартин Фаулер UML. Основи. 3-є видання. / Мартин Фаулер, Кендалл Скотт – Символ-Плюс, 2015. – 192 ст.
16. Методи розробки Web-сайтів [Електронний ресурс] – Режим доступу: <https://sites.google.com/site/tz5103voinovakateryna/metodi-rozrobki-web-sajtiv>
17. Глушаков С. В. Програмування WEB-сторінок / С. В. Глушаков, В. А. Жакін, Т. З. Хачиров. – Харків: Фоліо, 2004. – 390 с.
18. Кейт Джонс. DOM Scripting: Web Design with JavaScript and the Document Object Model. / К. Джонс 2005. — 368 с.
19. Edward Marteson Microsoft FrontPage: Essentials Explained, 2022. 20p.
20. Extending Visual Studio Code [Електронний ресурс] — Режим доступу: <https://code.visualstudio.com/docs/extensions/overview>
21. Види CMS — системи управління сайтами [Електронний ресурс] – Режим доступу: <https://www.avahost.ua/ua/vidi-cms-sistemi-upravlinnya-sajtami/>
22. Jennifer M., Elin W. Official Joomla! Book, The (Joomla! Press) 2nd Edition December 29, 2012. 495 p.

23. Що таке WordPress? [Електронний ресурс] – Режим доступу: <https://wpfan.com.ua/tak-scho-zh-take-wordpress/>

24. Афонін А. О. Особливості використання документо-орієнтованих баз даних на прикладі MongoDB / А. О. Афонін, Д. В. Зважій // Наукові записки НаУКМА. Комп'ютерні науки. - 2015. - Т. 177. - С. 113-117. - Режим доступу: http://nbuv.gov.ua/UJRN/NaUKMAkn_2015_177_22

25. Mongo DB [Електронний ресурс] – Режим доступу: <https://brander.ua/technologies/mongo-db>

26. UI Testing: A Detailed Guide [Електронний ресурс] – Режим доступу: <https://www.browserstack.com/guide/ui-testing-guide>

27. Д. Колесніков Як тестувати веб-сайт: основні етапи і поради [Електронний ресурс] – Режим доступу: <https://brainlab.com.ua/uk/blog-uk/yak-testuvati-veb-sayt-osnovn-etapi-poradi>

ДОДАТКИ

Додаток А – Технічне завдання

Міністерство освіти і науки України
Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії

УЗГОДЖЕНО
Завідувач Малинківського НВК
Турчанівська Н. П.
25 березня 2022 р.

ЗАТВЕРДЖУЮ
Завідувач кафедри ПЗ
Романюк О.Н.
25 березня 2022 р.

Технічне завдання
на бакалаврську дипломну роботу «Розробка веб-системи моніторингу
та підтримки навчально-виховного процесу у закладах освіти I ступеня»
за спеціальністю
121 – Інженерія програмного забезпечення

Керівник бакалаврської дипломної роботи:

_____ к.т.н., доц. В.В. Войтко

25 березня 2022 р.

Виконала:

_____ студентка гр. ЗПІ-186 Д.С.Лаба

25 березня 2022 р.

Вінниця – 2022 року

1. Найменування та галузь застосування

Бакалаврська дипломна робота: «Розробка веб-системи моніторингу та підтримки навчально-виховного процесу у закладах освіти І ступеня»

Галузь застосування – веб-технології.

2. Підстава для розробки.

Завдання на роботу, яке затверджене на засіданні кафедри програмного забезпечення – протокол № ____ від «____» _____ 2022 р.

3. Мета та призначення розробки.

Метою бакалаврської дипломної роботи є підвищення ефективності роботи закладу освіти І ступеня шляхом розробки спеціалізованої веб-системи з засобами моніторингу та підтримки навчально-виховного процесу в молодшій школі, що забезпечить ефективну взаємодію суб'єктів освітнього процесу.

Призначення роботи – моніторинг успішності учнів.

4. Вихідні дані для проведення НДР

1. Мартин Фаулер UML. Основи. 3-є видання. / Мартин Фаулер, Кендалл Скотт – Символ-Плюс, 2015. – 192 ст.
2. Web Development Recipes: навч. посіб. / Brian P. Hogan, Chris Warren, Pragmatic Bookshelf 2 edition, 2015. – 360 ст.
3. Кейт Джонс. DOM Scripting: Web Design with JavaScript and the Document Object Model. / К. Джонс 2005. — 368 с.
4. Афонін А. О. Особливості використання документо-орієнтованих баз даних на прикладі MongoDB / А. О. Афонін, Д. В. Зважій // Наукові записки НаУКМА. Комп'ютерні науки. - 2015. - Т. 177. - С. 113-117. - Режим доступу: http://nbuv.gov.ua/UJRN/NaUKMAkn_2015_177_22.

5. Технічні вимоги

Вхідні дані – інформація користувача, кількість учнів та вчителів; вихідні дані – графічний інтерфейс зі списком доступних учнів та оцінок.

6. Конструктивні вимоги.

Конструкція програми повинна відповідати естетичним вимогам, повинна бути зручною у використанні.

Графічна та текстова документація повинна відповідати діючим стандартам України.

7. Перелік технічної документації, що пред'являється по закінченню робіт:

- пояснювальна записка до бакалаврської дипломної роботи;
- технічне завдання;
- лістинги програми.

8. Вимоги до рівня уніфікації та стандартизації

При розробці програмних засобів слід дотримуватися уніфікації і ДСТУ.

Додаток Б – Акт впровадження

АКТ**про впровадження результатів бакалаврської дипломної роботи
«Веб-система для моніторингу та підтримки навчально-виховного
процесу у закладах освіти I ступеня»****студентки ВНТУ Лаби Діани Сергіївни**

Результати бакалаврської роботи «Розробка веб-системи моніторингу та підтримки навчально-виховного процесу у закладах освіти I ступеня» студентки ВНТУ Лаби Д. С., що полягають в розробці:

- методу моніторингу та підтримки навчально-виховного процесу, який передбачає розвинутий комплексний функціонал веб-ресурсу;
- моделей електронного журналу та алгоритмів роботи веб-системи;
- веб-системи, яка призначена для моніторингу успішності учнів навчально-виховного закладу.

впроваджені в «Малинківський навчально-виховний комплекс I ст.-сад.».

Завідувач Малинківського НВК

Турчанівчська Н.П

10.06.2022

Додаток В – Протокол перевірки на плагіат

ПРОТОКОЛ
ПЕРЕВІРКИ БАКАЛАВРСЬКОЇ ДИПЛОМНОЇ РОБОТИ
НА НАЯВНІСТЬ ТЕКСТОВИХ ЗАПОЗИЧЕНЬ

Назва роботи: Розробка веб-системи моніторингу та підтримки навчально-виховного процесу у закладах освіти I ступеня

Тип роботи: БДР

Підрозділ : кафедра програмного забезпечення, ФІТКІ

Науковий керівник: к.т.н., доц. каф. ПЗ Войтко В.В.

Оригінальність	95.3%
Схожість	4.7%

Аналіз звіту подібності

■ **Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.**

Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її автора. Роботу направити на доопрацювання.

Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Особа, відповідальна за перевірку _____ Черноволик Г. О.

Ознайомлені з повним звітом подібності, який був згенерований системою Unichек

Автор роботи _____

Лаба Д. С.

Керівник роботи _____

Войтко В.В.

Додаток Г – Лістинг програми

Серверна частина веб-системи

Index.js

```
const express = require('express')
const config = require('config')
const mongoose = require('mongoose')

const app = express()

app.use(express.json({ extended: true }))

app.use('/api/auth', require('./routes/auth'))
app.use('/api/school', require('./routes/school'))

const PORT = config.get('port') || 5000

async function start() {
  try {
    await mongoose.connect(config.get('mongoUrl'), {

    })
    app.listen(PORT, () => {
      console.log(`App has been started on port ${PORT}...`);
    })
  } catch (e) {
    console.log('Server Error', e.message)
    process.exit(1)
  }
}

start()
```

Progress.js


```

const {Schema, model, Types} = require('mongoose')

const schema = new Schema({
  classroom: {type: Number, required: true},
  teacher_id: {type: Number, required: true},
  teacher_name: {type: String, required: true},
  students: [{
    name: String,
    stud_id: Number,
    marks: [{
      sub_id: Number,
      mark: String
    }],
  }],
  subjects: [{
    name: String,
    sub_id: Number
  }],
  class_id: {type: Number, required: true},
})

module.exports = model('Progress', schema);

```

Statement.js

```

const {Schema, model, Types} = require('mongoose')

const schema = new Schema({
  number: {type: String, required: true},
  name: {type: String, required: true},
  sex: {type: String, required: true},
  birth_date: {type: String, required: true},
  parents: [{type: String}],
  phone: {type: String, required: true},

```

```

    address: {type: String, required: true},
    teacher_id: {type: Number, required: true}
  })

```

```

module.exports = model('Statement', schema);

```

User.js

```

const {Schema, model, Types} = require('mongoose')

```

```

const schema = new Schema({
  email: {type: String, required: true, unique: true},
  password: {type: String, required: true},
  user_name: {type: String, required: true},
  user_id: {type: Number}
})

```

```

module.exports = model('User', schema);

```

Auth.js

```

const {Router} = require('express')

```

```

const User = require('../models/User')

```

```

const router = Router()

```

```

router.post(

```

```

  '/login',

```

```

  async (req, res) => {

```

```

    try {

```

```

      const {email, password} = req.body;

```

```

      const user = await User.findOne({ email })

```

```

      if (!user) {

```

```

        return res.status(400).json({ message: 'Неправильний
пароль або емейл'})

```

```

    }

    if (user.password !== password) {
        return res.status(400).json({ message: 'Неправильний
пароль або емейл'})
    }

    res.json({userId: user.user_id})

} catch (e) {
    res.status(500).json({ message: 'Something went wrong,
try again' })
}
})
module.exports = router

```

School.js

```

const {Router} = require('express')
const Statement = require('../models/Statement')
const Progress = require('../models/Progress')
const router = Router()

router.get(
    '/statement/:id',
    async (req, res) => {
        try {
            const statementId = req.params.id;

            const statementData = await Statement.find({ teacher_id:
statementId});

            if (!statementData) {
                return res.status(400).json({ message: 'not found'})
            }

```

```
        res.json({data: statementData})

    } catch (e) {
        console.log(e);
        res.status(500).json({ message: 'Something went wrong,
try again' })
    }
})

router.get(
    '/progress/:id',
    async (req, res) => {
        try {
            const progressId = req.params.id;

            const progressData = await Progress.findOne({ teacher_id:
progressId});

            if (!progressData) {
                return res.status(400).json({ message: 'not found'})
            }

            res.json({data: progressData})

        } catch (e) {
            console.log(e);
            res.status(500).json({ message: 'Something went wrong,
try again' })
        }
    })

module.exports = router
```

Клієнтська частина веб-системи

Index.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <link rel="icon" href="%PUBLIC_URL%/favicon.ico" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <meta name="theme-color" content="#000000" />
    <meta
      name="description"
      content="Web site created using create-react-app"
    />
    <link rel="apple-touch-icon" href="%PUBLIC_URL%/logo192.png" />

    <link rel="manifest" href="%PUBLIC_URL%/manifest.json" />

    <link
      href="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/css/materialize.min.css"
      rel="stylesheet"
    />

    <script
      src="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/js/materialize.min.js"></script>
    <link rel="preconnect" href="https://fonts.googleapis.com">
    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
    <link
      href="https://fonts.googleapis.com/css2?family=Courgette&family=Lora:wght@400;700&family=Montserrat:ital,wght@0,400;0,700;1,400&display=swap"
      rel="stylesheet">

    <title>React App</title>
  </head>
```

```

<body>
  <noscript>You need to enable JavaScript to run this app.</noscript>
  <div id="root"></div>

</body>
</html>

```

App.js

```

import React from "react";
import Navbar from "./components/navbar";
import { Routes, Route, useLocation } from "react-router-dom"
import Header from "./components/header";
import AboutPage from "./pages/about";
import StudentsPage from "./pages/students";
import ParentsPage from "./pages/parents";
import SchoolLifePage from "./pages/school-life";
import DocsPage from "./pages/docs";
import ProgressPage from "./pages/progress";
import LoginPage from "./pages/login";
import Footer from "./components/footer";
function App() {

  const location = useLocation()

  if (location.pathname === '/login') {
    return (
      <Routes>
        <Route path="/login" element={<LoginPage />} />
      </Routes>
    )
  }
  return (
    <div className="App">
      <Header />

```

```

    <Navbar />
    <div className="">
      <Routes>
        <Route path="/" element={<AboutPage />} />
        <Route path="/students" element={<StudentsPage />} />
        <Route path="/parents" element={<ParentsPage />} />
        <Route path="/school-life" element={<SchoolLifePage />} />
        <Route path="/docs" element={<DocsPage />} />
        <Route path="/progress" element={<ProgressPage />} />
      </Routes>
    </div>
    <Footer />
  </div>
);
}

export default App;

```

Index.js

```

import React from 'react';
import ReactDOM from 'react-dom';
import { BrowserRouter } from 'react-router-dom';
import './index.css';
import App from './App';

ReactDOM.render(
  <React.StrictMode>
    <BrowserRouter>
      <App />
    </BrowserRouter>
  </React.StrictMode>,
  document.getElementById('root')
);

```

Progress-table.js

```

import React from 'react';

const ProgressTable = ({data}) => {

  const tableHeaders = data.subjects.map((el, idx) => {
    return (
      <th key={`-${el}-${idx}`} >
        {el.name}
      </th>
    )
  })

  const tableBody = data.students.map((el, idx) => {
    const bodyData = []
    bodyData.push(
      <td key={`-${idx}`} >{el.name}</td>
    );

    data.subjects.map((s, i) => {
      for (const mark of el.marks) {
        if (s.sub_id === mark.sub_id) {
          bodyData.push(
            <td key={`-${mark.mark}-${i}`} >{mark.mark}</td>
          );
        }
      }
    })
  })

  return (
    <tr key={`-${el}-${idx}`} >
      <td>{idx+1}</td>
      {bodyData}
    </tr>
  )
}

```



```

    )
  })

  return (
    <table className='striped'>
      <thead>
        <tr>
          <th>№</th>
          <th>Учні</th>
          {tableHeaders}
        </tr>
      </thead>

      <tbody>
        {tableBody}
      </tbody>
    </table>
  );
}

```

```
export default ProgressTable
```

Progress.js

```

import React, {useState, useEffect} from 'react';
import StatementTable from '../components/statement-table';
import ProgressTable from '../components/progress-table';

const ProgressPage = () => {

  const userId = JSON.parse(localStorage.getItem('adminData')).userId;

  const [statementData, setStatementData] = useState(null);
  const [progressData, setProgressData] = useState(null);
  useEffect(() => {

```

```

fetch(`/api/school/statement/${userId}`)
  .then(res => res.json())
  .then(json => {
    setStatementData(json.data)
  })

fetch(`/api/school/progress/${userId}`)
  .then(res => res.json())
  .then(json => {
    setProgressData(json.data)
  })
}, [])

if (!progressData || !statementData) {
  return(
    <p>
      loading
    </p>
  )
}

const st = statementData.length ? <StatementTable data={statementData}
titles=[["№", "Номер", "ПІБ", "Стать", "Дата народження", "ПІБ батьків",
"Телефон", "Адреса", ""]] /> : null

return (
  <div className='my_table'>
    {st}
    <h3>Клас {progressData.classroom}</h3>
    <h5>{progressData.teacher_name}</h5>
    <ProgressTable data={progressData}/>
  </div>

```

```

    );
}

```

```
export default ProgressPage
```

Login.js

```
import React, {useState, useEffect} from 'react';
```

```
import {useNavigate} from 'react-router-dom';
```

```
const LoginPage = () => {
```

```

    const [form, setForm] = useState({
      email: '',
      password: ''
    })

```

```
const [inputClass, setInputClass] = useState('validate');
```

```
const navigate = useNavigate()
```

```
useEffect(() => {
```

```
}, [inputClass])
```

```
const changeHandler = e => {
```

```
  setForm({ ...form, [e.target.name]: e.target.value })
```

```
}
```

```
const onLogin = async (e) => {
```

```
  e.preventDefault();
```

```
  try {
```

```
    const response = await fetch('/api/auth/login', {
```

```
      method: "POST",
```

```
      body: JSON.stringify(form),
```

```
        headers: new Headers(  
            {  
                "Content-Type": "application/json",  
                "Accept": "application/json"  
            }  
        ),  
    })  
  
    const data = await response.json()  
  
    if (!response.ok) {  
        alert(data.message)  
        setInputClass(inputClass + " invalid")  
        return  
    }  
  
    setInputClass("validate");  
    localStorage.setItem("adminData", JSON.stringify(data))  
    navigate('/');  
  
    } catch (e) {  
        console.error(e)  
    }  
}  
  
const onCancel = () => {  
    navigate('/');  
}  
  
return (  
    <div className='my_login'>  
        <div className="row login_content">
```

```

<form className="col s6" >
  <div className="row">
    <div className="input-field col s12">
      <input id="email"
        type="email"
        name="email"
        className={inputClass}
        onChange={changeHandler}
        value={form.email}/>
      <label htmlFor="email">Email</label>
    </div>
  </div>
  <div className="row">
    <div className="input-field col s12">
      <input id="password"
        type="password"
        name="password"
        className={inputClass}
        onChange={changeHandler}
        value={form.password}/>
      <label htmlFor="password">Password</label>
    </div>
  </div>
  <button className="waves-effect waves-light btn"
    onClick={onLogin}>Війти</button>
  <button className="waves-effect waves-light btn red
darken-3 my_cancel"
    onClick={onCancel}>Повернутись</button>
</form>
</div>
</div>
);
}

```

```
export default LoginPage
```

G.js

```
import React, {useState} from "react";
```

```
import "./index.css"
```

```
const G = () => {
```

```
    const n = ["Галяновська Анна Іванівна", "Кушнір Олександр Сергійович",
"Кушнір Володимир Сергійович", "Польгуль Богдан Ігорович", "Маринич Анна
Петрівна"]
```

```
    const s = ["Українська мова і літературне навчання", "Англійська мова",
"Математика", "Я досліджую світ (інтегрований курс)", "Інформатика",
"Музичне мистецтво", "Образотворче мистецтво", "Фізкультура"]
```

```
    const onGen = async () => {
```

```
        const clr = {
```

```
            classroom: 4,
```

```
            teacher_id: 3,
```

```
            teacher_name: 'Лаба Світлана Анатоліївна',
```

```
            subjects: [],
```

```
            class_id: 3,
```

```
            students: []
```

```
        }
```

```
        for (let index = 0; index < n.length; index++) {
```

```
            let obj = {
```

```
                name: n[index],
```

```
                stud_id: index,
```

```
                marks: [],
```

```
            }
```

```
            for (let i = 0; i < s.length; i++) {
```

```
                let mark;
```

```
                let sub_id = i;
```

```
                let num = Math.floor(Math.random() * 2);
```

```
        if (num === 0) {
            mark = "добре"
        } else {
            mark = "відмінно"
        }
        obj.marks.push({
            sub_id,
            mark
        })
    }
    clr.students.push(obj)
}

s.forEach((element, i) => {
    clr.subjects.push({
        name: element,
        sub_id: i
    })
});

const response = await fetch('/api/school/generate', {
    method: "POST",
    body: JSON.stringify(clr),
    headers: new Headers(
        {
            "Content-Type": "application/json",
            "Accept": "application/json"
        }
    ),
})

const data = await response.json()
}
return (
```

```

    <div>
      <button onClick={onGen}>go</button>
    </div>
  );
}

```

```
export default G
```

Statement-table.js

```

import React from 'react';

const StatementTable = ({data, titles}) => {

  const tableHeaders = titles.map((el, idx) => {
    return (
      <th key={` ${el}-${idx}`} >
        {el}
      </th>
    )
  })

  const tableBody = data.map((el, idx) => {

    const bodyData = []

    for (const k in el) {
      if (!k.includes('id')) {
        if (k === "parents") {
          const parents = el.parents.map((par, i) => {
            if (el.parents.length-1 === i) {
              return <span key={par}>{par}</span>
            }
          })
          return (

```



```

        <span key={par}>
            {par}
            <br/>
        </span>
    )
    })
    bodyData.push(<td key={k}>{parents}</td>)
    continue
    }
    bodyData.push(
        <td key={k}>{el[k]}</td>
    )
    }
}

return (
    <tr key={` ${el}-${idx}` }>
        <td>{idx+1}</td>
        {bodyData}
    </tr>
)
})

return (
    <table className='striped'>
    <thead>
        <tr>
            {tableHeaders}
        </tr>
    </thead>

    <tbody>
        {tableBody}
    </tbody>
)

```

```
        </table>  
    );  
}
```

```
export default StatementTable
```

ГРАФІЧНА ЧАСТИНА
РОЗРОБКА ВЕБ-СИСТЕМИ МОНІТОРИНГУ ТА ПІДТРИМКИ НАВЧАЛЬНО-
ВИХОВНОГО ПРОЦЕСУ У ЗАКЛАДАХ ОСВІТИ І СТУПЕНЯ

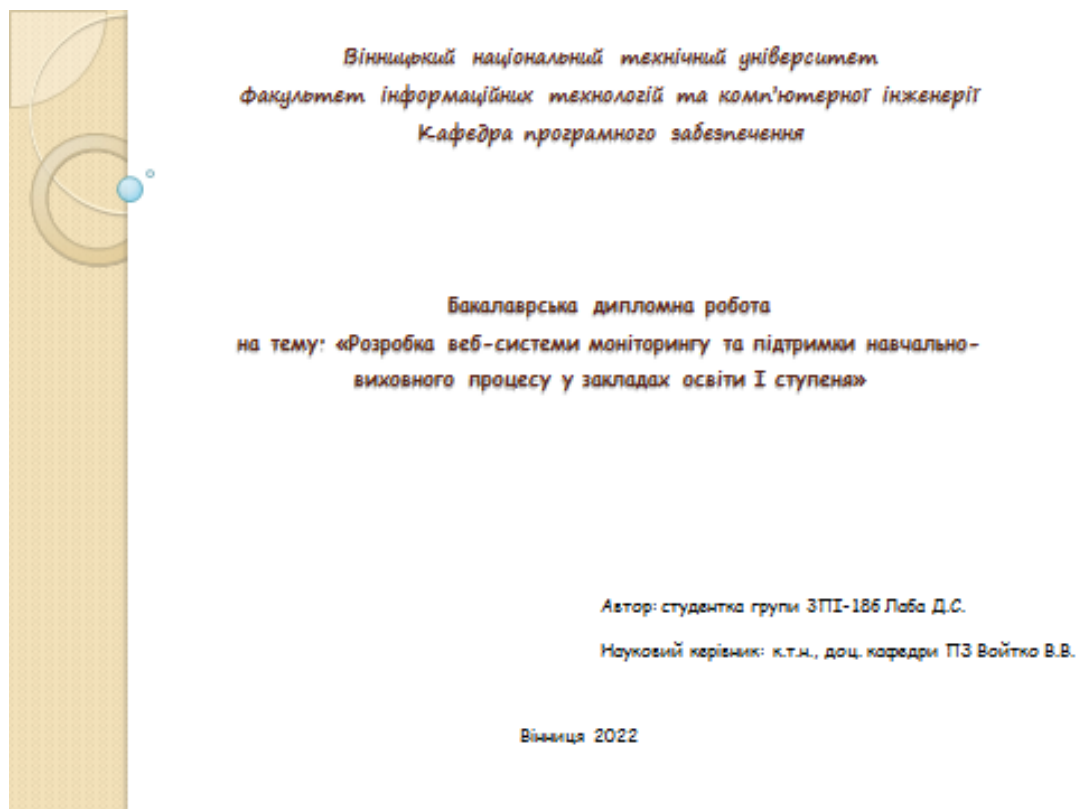


Рисунок Д.1 – Титульний слайд

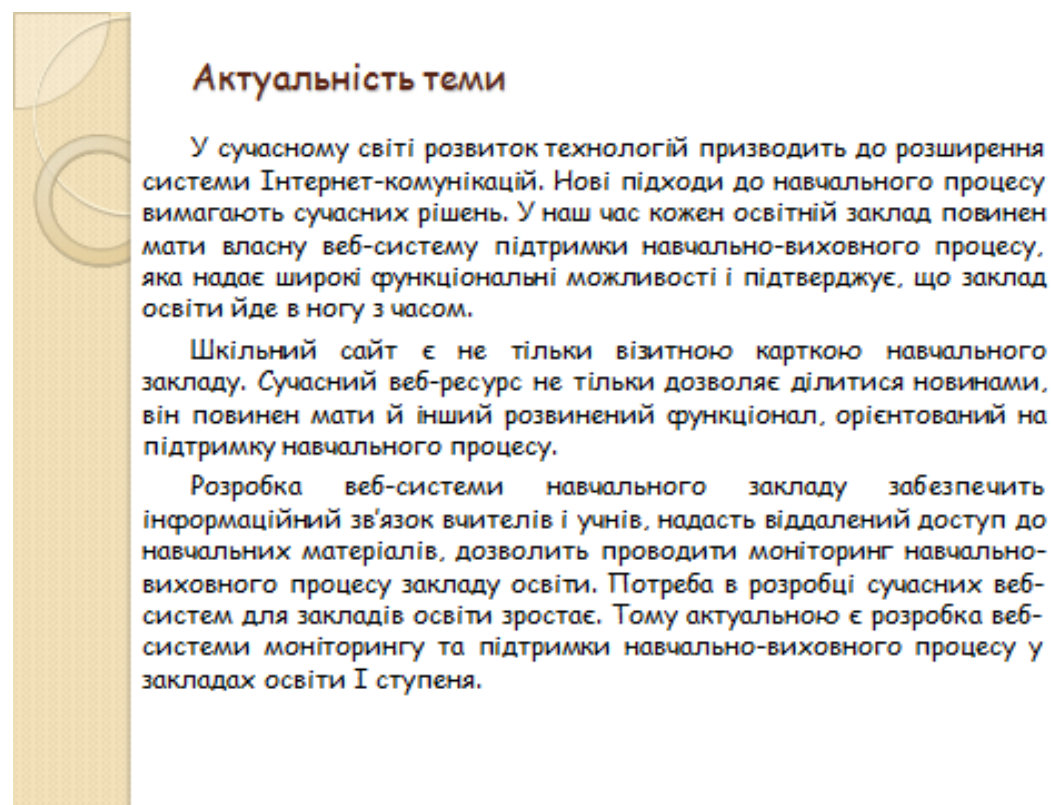


Рисунок Д.2 – Актуальність теми

Мета, об'єкт та предмет дослідження

Метою бакалаврської дипломної роботи є підвищення ефективності роботи закладу освіти I ступеня шляхом розробки спеціалізованої веб-системи з засобами моніторингу та підтримки навчально-виховного процесу в молодшій школі, що забезпечить ефективну взаємодію суб'єктів освітнього процесу.

Об'єктом дослідження є процес розробки веб-системи моніторингу та підтримки навчально-виховного процесу у закладах освіти I ступеня.

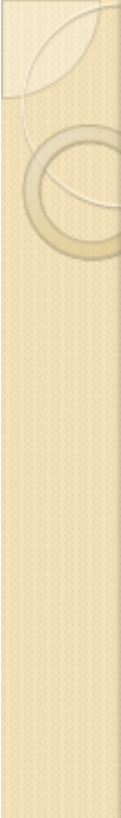
Предметом дослідження є засоби реалізації веб-системи закладу освіти I ступеня.

Рисунок Д.3 – Мета, об'єкт та предмет дослідження

Основними задачами є:

- розробити метод моніторингу та підтримки навчально-виховного процесу у закладах освіти з використанням спеціалізованої веб-системи;
- розробити модель роботи веб-системи;
- проаналізувати інформаційне забезпечення сайту;
- розробити програмні засоби реалізації веб-системи моніторингу та підтримки навчально-виховного процесу;
- розробити зручний та зрозумілий графічний інтерфейс веб-ресурсу;
- провести тестування веб-системи відповідно до поставлених задач.

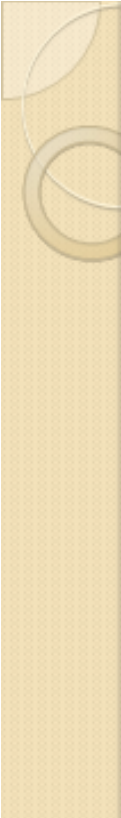
Рисунок Д.4 – Задачі дослідження



Наукова новизна

- ▶ Подальшого розвитку отримав метод моніторингу та підтримки навчально-виховного процесу у закладах освіти, який, на відміну від існуючих, базується на використанні комплексного функціоналу моніторингу успішності учнів, наповнення і використання навчально-методичних матеріалів і підтримки виховного процесу, що дозволяє підвищити ефективність освітнього процесу в умовах дистанційного доступу.
- ▶ Подальшого розвитку отримала модель роботи веб-системи закладу освіти I ступеня, яка, на відміну від існуючих, орієнтована на покращенні навчально-виховного процесу шляхом реалізації комплексного функціоналу та розвиненої системи зв'язків для забезпечення взаємодії суб'єктів освітнього процесу, що дозволяє підвищити якість навчання у молодшій школі.

Рисунок Д. 5 – Наукова новизна



Практична цінність одержаних результатів

Практична цінність полягає у кінцевій реалізації інформаційної веб-системи навчального закладу I ступеня, що дозволить ефективно підтримувати навчально-виховний процес у закладі освіти.

Рисунок Д.6 – Практична цінність одержаних результатів

Аналоги

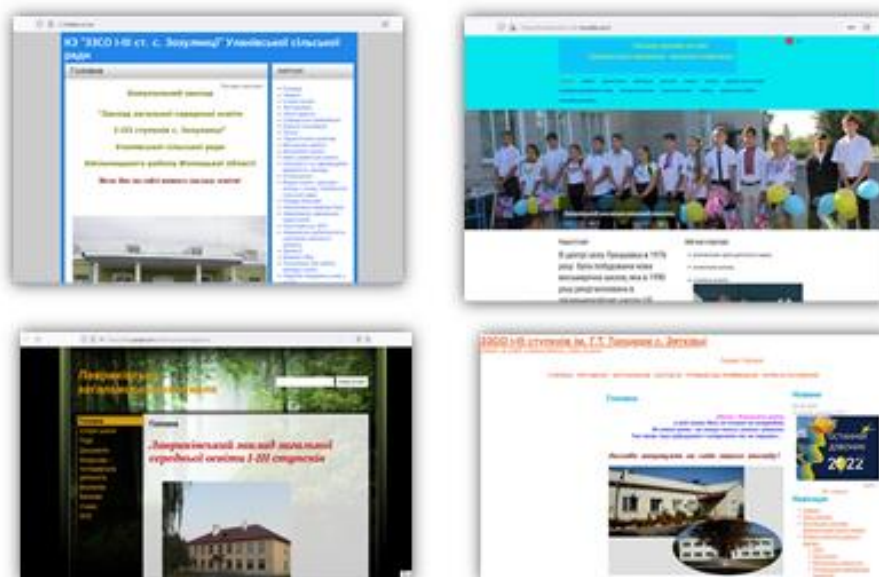


Рисунок Д.7 – Аналоги

Порівняння аналогів

Критерій	Сайт Лазарківського закладу загальної середньої освіти	Сайт Лукашківського навчально-виховного комплексу	Сайт Закладу загальної середньої освіти с. Золудинці	Сайт ЗСО І-ІІІ ступенів ім. Г.Т. Танцюри с. Затишці	Власна веб-система
Зручна навігація	-	+	+	+	+
Авторизація	-	-	-	-	+
Новини	+	+	+	+	+
Читабельність тексту	+	-	+	+	+
Зрозуміле меню	+	+	+	+	+
Загальна оцінка	60%	60%	80%	80%	100%

Рисунок Д.8 – Порівняльний аналіз аналогів

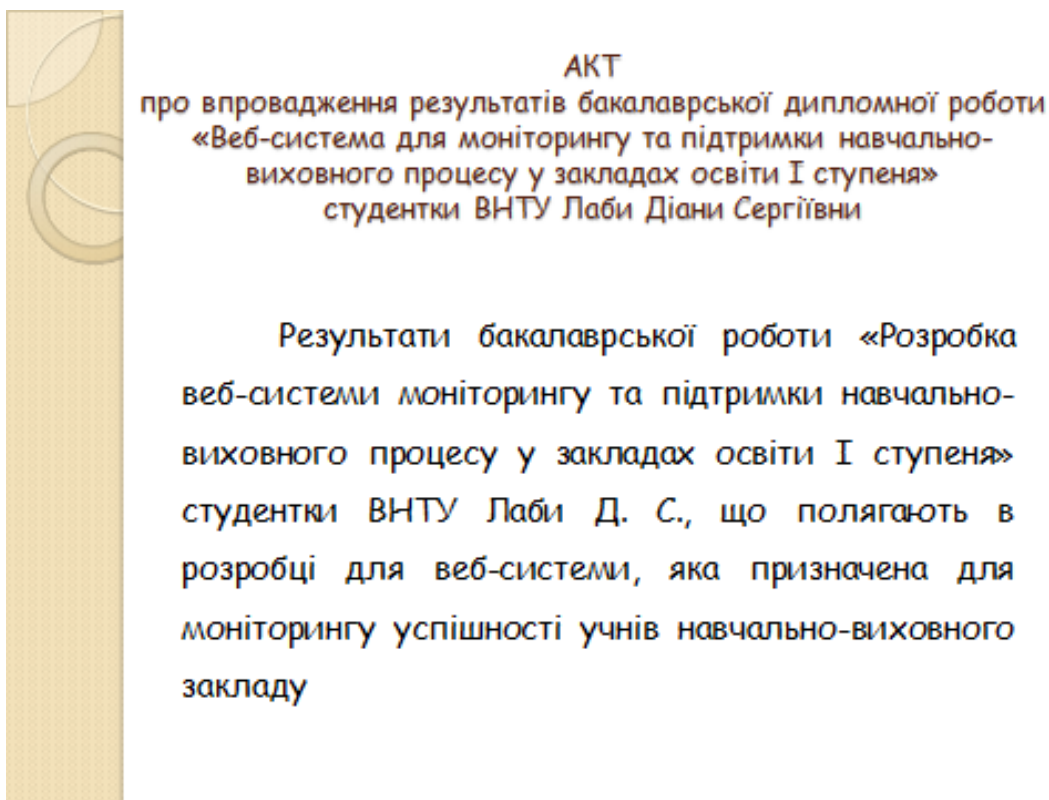


Рисунок Д.9 – Акт про впровадження наукової роботи



Рисунок Д.10 – Схема реалізації методу моніторингу та підтримки навчально-виховного процесу в закладах освіти

Модель роботи веб-системи

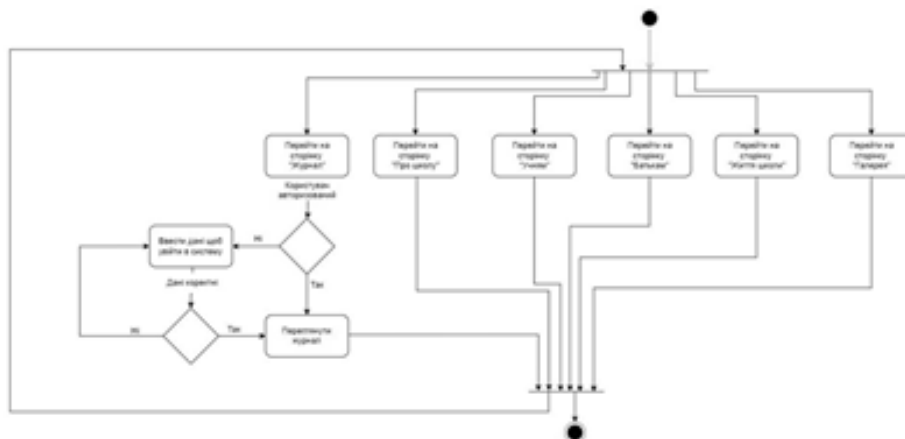


Рисунок Д.11 – Модель роботи веб-системи

Блок-схема загального алгоритму програми

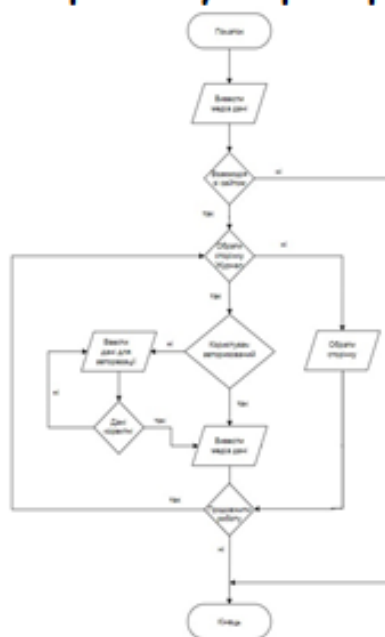


Рисунок Д.12 – Блок-схема алгоритму роботи електронного журналу



Рисунок Д.13 – Блок-схема алгоритму роботи електронного журналу

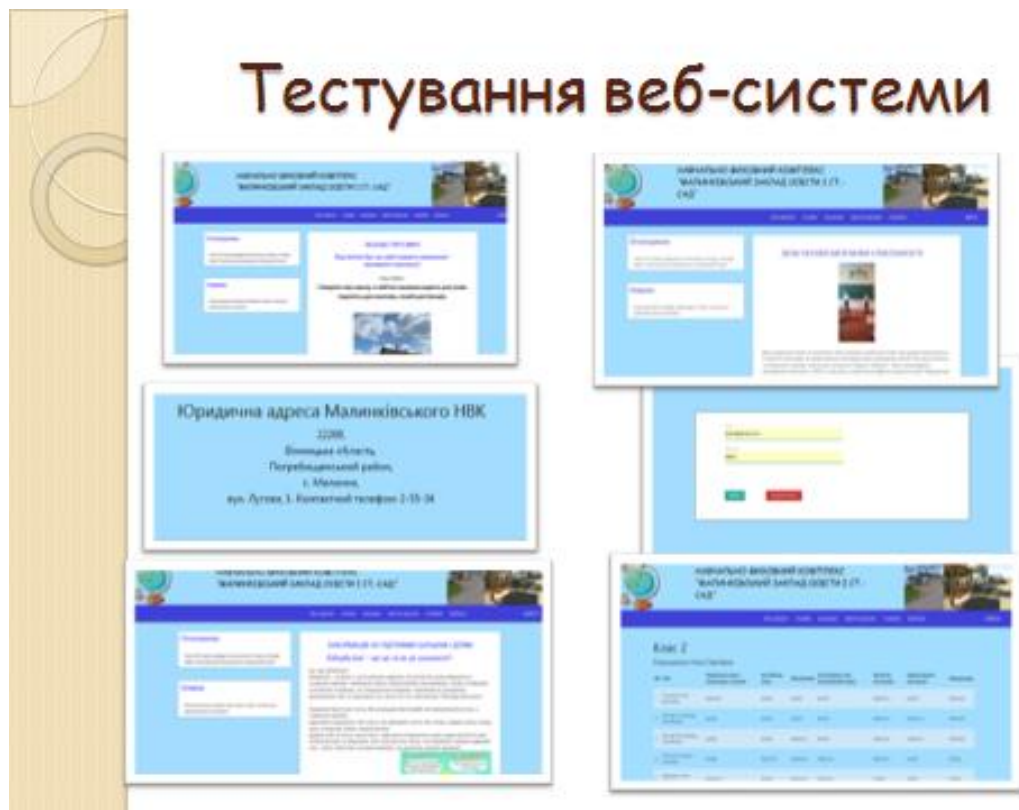


Рисунок Д.14 – Тестування веб-системи

Апробація матеріалів бакалаврської дипломної роботи. Основні положення бакалаврської дипломної роботи доповідалися та обговорювалися на ІІ науково-технічній конференції факультету інформаційних технологій та комп'ютерної інженерії (2022) ВНТУ.

Публікації. Войтко В.В. Особливості розробки веб-сайту навчально-виховного комплексу "Малинківський заклад освіти І ст.- САД " / В.В. Войтко, Л.М.Круподьорова, А. В. Денисюк, О. В. Гаврилюк, Н. Є. Барчук, Д. С. Лаба // Матеріали ІІ Науково-технічна конференція факультету інформаційних технологій та комп'ютерної інженерії (2022) ВНТУ, Факультет інформаційних технологій та комп'ютерної інженерії, Секція програмного забезпечення. [Електронний ресурс] - Режим доступу до ресурсу: <https://conferences.vntu.edu.ua/index.php/all-fitki/all-fitki-2022/paper/view/15097/12733>

Рисунок Д.15 – Апробація результатів роботи та публікації

Висновки

У бакалаврській дипломній роботі було розроблено веб-систему для моніторингу та підтримки навчально-виховного комплексу. Для розробки було використано середовище програмної розробки Microsoft Visual Code.

Під час виконання бакалаврської дипломної роботи було проаналізовано стан задачі, розглянуто основні аналоги програмного продукту, визначено їхні переваги і недоліки та порівняно з власним програмним продуктом. Опіраючись на результати порівняння, було сформульовано основні задачі бакалаврської дипломної роботи.

Під час аналізу технологій розробки було обґрунтовано вибір мови програмування JavaScript. Серверна частина розроблена за допомогою Node.js. А для зберігання даних використано базу даних MongoDB.

Рисунок Д.16 – Висновки (перша частина)

Висновки

У бакалаврській дипломній роботі було зроблено:

- ❖ розроблено метод моніторингу та підтримки навчально-виховного процесу у закладах освіти з використанням спеціалізованої веб-системи;
- ❖ розроблено модель роботи веб-системи;
- ❖ проаналізовано інформаційне забезпечення сайту;
- ❖ розроблено програмні засоби реалізації веб-системи моніторингу та підтримки навчально-виховного процесу;
- ❖ проведено тестування роботи веб-системи.

Тестування програми показало цілковиту працездатність програмного продукту та відповідність поставленому технічному завданню. Також було розроблено інструкцію користувача.

Рисунок Д.17 – Висновки (друга частина)