

Галицький коледж імені В'ячеслава Чорновола  
відділення комп'ютерних та видавничих технологій  
циклова комісія інформатики та комп'ютерних дисциплін

ДОПУСТИТИ ДО ЗАХИСТУ  
Завідувач відділенням  
комп'ютерних та видавничих  
технологій

Чубей О.О. / \_\_\_\_\_ /

підпис

« \_\_\_\_ » \_\_\_\_\_

2021 р.

## ПОЯСНЮВАЛЬНА ЗАПИСКА

до дипломного проєкту  
освітньо-кваліфікаційного рівня «молодший спеціаліст»  
зі спеціальності 122 «Комп'ютерні науки»

на тему: «Сайт Нижчелуб'янківської ЗОШ І-ІІІ ступенів»

Студент групи К-47

Риндич Б.А.

\_\_\_\_\_

(підпис)

Керівник проєкту

Кульчинська Н.З.

\_\_\_\_\_

(підпис)

Консультанти:

з техніко-економічного  
обґрунтування

Меленчук Л.І.

\_\_\_\_\_

(підпис)

нормоконтролер

Гавришків Н.Г.

\_\_\_\_\_

(підпис)

Тернопіль – 2021

Галицький коледж імені В'ячеслава Чорновола  
відділення комп'ютерних та видавничих технологій  
циклова комісія інформатики та комп'ютерних дисциплін

ЗАТВЕРДЖУЮ

Завідувач відділенням  
комп'ютерних та видавничих  
технологій

Чубей О.О. / \_\_\_\_\_ /  
підпис

« \_\_\_\_ » \_\_\_\_\_ 2020 р.

ЗАВДАННЯ

на дипломне проєктування  
на здобуття освітньо-кваліфікаційного рівня «молодший спеціаліст»

студенту \_\_\_\_\_  
(прізвище, ім'я та по-батькові студента)

1. Тема проєкту \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

затверджена наказом по коледжу від “ \_\_\_\_ ” \_\_\_\_\_ 2020 р., № \_\_\_\_

2. Термін здачі студентом завершеного проєкту “ \_\_\_\_ ” \_\_\_\_\_ 2021 р.

3. Вихідні дані до проєкту \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

4. Перелік питань, які повинні бути розроблені в проєкті:

а) основна частина \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

б) техніко-економічного обґрунтування \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

5. Перелік графічного матеріалу \_\_\_\_\_  
\_\_\_\_\_

6. Консультанти проєкту: \_\_\_\_\_

| Розділ                               | Консультанти  | Підпис, дата    |                   |
|--------------------------------------|---|-----------------|-------------------|
|                                      |   | Завдання видано | Завдання прийнято |
| з техніко-економічного обґрунтування | _____<br>(вчена ступень, звання П.І.Б.<br>консультанта) |                 |                   |

## КАЛЕНДАРНИЙ ПЛАН дипломного проєктування

| №<br>п/п | Найменування етапу  | Терміни     |             |
|----------|---|-------------|-------------|
|          |   | початку     | завершення  |
| 1.       | Вибір теми, ознайомлення з вимогами до дипломного проєктування.   | 16.11.20 р. | 30.11.20 р. |
| 2.       | Огляд типових рішень та написання відповідного розділу ПЗ   | 01.12.20 р. | 26.01.21 р. |
| 3.       | Дослідження технологій реалізації та написання відповідного розділу ПЗ  | 27.01.21 р. | 15.02.21 р. |
| 4.       | Розробка функціональних вимог до проєкту та робота над структурою програмного продукту. Написання відповідного розділу ПЗ | 15.02.21 р. | 02.03.21 р. |
| 5.       | Встановлення на налаштування середовища реалізації та написання відповідного розділу ПЗ                                   | 02.03.21 р. | 16.03.21 р. |
| 6.       | Проєктування програмного засобу (функціоналу, інтерфейсу, бази даних продукту) та написання відповідного розділу ПЗ       | 16.03.21 р. | 16.04.21 р. |
| 7.       | Реалізація та налаштування програмного засобу та написання відповідного розділу ПЗ  | 17.04.21 р. | 03.05.21 р. |
| 8.       | Доопрацювання модулів   | 03.05.21 р. | 17.05.21 р. |
| 9.       | Тестування на налагодження програмного продукту та написання відповідного розділу ПЗ                                      | 18.05.21 р. | 04.06.21 р. |
| 10.      | Опрацювання економічного розділу дипломного проєкту та оформлення спеціального розділу                                    | 17.05.21 р. | 18.06.21 р. |
| 11.      | Робота над оформленням пояснювальної записки  | 04.06.21 р. | 11.06.21 р. |
| 12.      | Попередній захист дипломного проєкту, доопрацювання   | 11.06.21 р. | 11.06.21 р. |
| 13.      | Підготовка до захисту дипломного проєкту  | 18.06.21 р. | 24.06.21 р. |
| 14.      | Захист дипломного проєкту   | 24.06.21 р. | 24.06.21 р. |

7. Дата видачі завдання “\_\_\_” \_\_\_\_\_ 2020 р.

Керівник \_\_\_\_\_/

Завдання прийняв до виконання \_\_\_\_\_/

## Реферат

Дипломний проєкт. Тема: «Сайт Нище Луб'янківської ЗОШ І-ІІІ ступенів». 84 сторінок, 58 рисунків, 1 таблиця, 8 джерел, 1 додато,.

Мета проєкту – розробка веб-сайту для Нище Луб'янківської ЗОШ.

Завданням проєкту є розробка сайту для школи який буде розміщувати в собі всі актуальні новини

В дипломному проєкті досліджено предметну область, проєктовано структуру та інтерфейс сайту, проаналізовано засоби його створення, відображено етапи реалізації та результати тестування, розглянуто питання економічної частини проєкту.

Результат – запроєктований та розроблений веб-сайт для школи який має зручний інтерфейс.

Для реалізації сайту було обрано платформу для розробки Node. База даних mongoDB, фронт частина написана на React.js. Програмування відбувається в середовищі Visual Studio Code.

WEB-САЙТ, ЗОШ, САЙТ ШКОЛИ, НОВИНИ, NPM, MONGODB, REACT, JAVASCRIPT, NODE, ШКОЛА.

## Abstract

Diploma project. Topic: "Site of the Nyzhchi Lubianky Secondary School of 1-3 grades". 84 pages, 58 figures, 1 table, 8 sources, 1 appendix ,.

The objective of the project is to develop a website for Nishche Lubyankivska Secondary School.

The purpose of the project is to develop a website for the school that will contain all the latest news

The diploma project explores the subject area, designed the structure and interface of the site, analyzed the means of its creation, reflected the stages of implementation and test results, considered the economic part of the project.

The result is a designed and developed website for the school that has a user-friendly interface.

A platform for Node development was chosen to implement the site. The mongoDB database, the front part is written in React.js. Programming takes place in the Visual Studio Code environment.

WEB SITE, SCHOOL, SCHOOL SITE, NEWS, NPM, MONGODB, REACT, JAVASCRIPT, NODE, SCHOOL.

## ЗМІСТ

|  |    |
|--|----|
| Вступ .....  | 7  |
| 1 Аналіз існуючих рішень та постановка завдання.....       | 8  |
| 1.1 Обґрунтування доцільності створення web-сайту .....    | 8  |
| 1.2 Огляд існуючих рішень .....                            | 8  |
| 1.3 Постановка задачі .....                                | 16 |
| 2 Проєктування web-сайту .....                             | 18 |
| 2.1 Проєктування структури web-сайту.....                  | 18 |
| 2.2 Проєктування користувацького інтерфейсу .....          | 19 |
| 2.3 Аналіз підходів до реалізації сайту .....              | 21 |
| 3 Реалізація та тестування.....                            | 22 |
| 3.1 Опис засобів реалізації .....                          | 22 |
| 3.2 Створення проєкту за допомогою платформи Node JS ..... | 28 |
| 3.3 Опис та реалізація власного дизайну для сайту .....    | 32 |
| 3.4 Опис додавання контенту для сайту .....                | 35 |
| 4 Техніко-економічне обґрунтування .....                   | 52 |
| 4.1 Аналіз ринку .....                                     | 52 |
| 4.2 Розрахунок витрат на проєктування .....                | 52 |
| 4.3 Обґрунтування необхідності розробки.....               | 54 |
| Висновки .....   | 55 |
| Перелік джерел посилань .....                              | 56 |
| Додатки .....  | 57 |

|            |      |                  |        |      |  |  |  |               |      |         |
|------------|------|------------------|--------|------|--|--|--|---------------|------|---------|
|            |      |                  |        |      | ДП. КН 21.466.11.000 ПЗ                            |  |  |               |      |         |
|            |      |                  |        |      |  |  |  |               |      |         |
| Зм.        | Арк. | № докум.         | Підпис | Дата | Сайт Нижчелуб'янківської<br><br>ЗОШ І-ІІІ ступенів |  |  | Літ.          | Арк. | Аркушів |
| Розроб.    |      | Риндич Б.А.      |        |      |  |  |  |               |      |         |
| Перев.     |      | Кульчинська Н.З. |        |      |  |  |  |               | 5    | 84      |
| Рецензент. |      | Кузик В.М.       |        |      |  |  |  | ГК. КВТ. К-47 |      |         |
| Н. Контр.  |      | Гавришків Н.Г.   |        |      |  |  |  |               |      |         |
| Зав. від.  |      | Чубей О.О.       |        |      |  |  |  |               |      |         |

## СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

WEB – World Wide Web

DB – Database

ПК – персональний комп'ютер

ODM – Object Data Modelling

ORM – Object-relational mapping

HTML – HyperText Markup Language

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 6    |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

## ВСТУП

Шкільний веб-сайт дуже важливий для створення першого хорошого враження, оскільки часто це перший контакт людей зі школою. Тому шкільний веб-сайт впливає на загальний імідж та репутацію школи, що може призвести до збільшення кількості відвідувачів. Наявність хорошого шкільного веб-сайту також дуже важливо для підвищення іміджу закладу для своїх постійних користувачів, включаючи вчителів, учнів та батьків.

Загальний вигляд веб-сайту – це перше, що помічають користувачі, відвідуючи веб-сайт. Це включає макет, кольори, зображення та шрифти, які використовуються для створення веб-сайту. Дизайн веб-сайту та хороший вміст дуже важливі, однак, якщо відвідувачі не можуть легко знайти інформацію, яку вони шукають, вони будуть розчаровані, і це негативно вплине на перший погляд користувачів.

Це особливо важливо, коли батьки вивчають школи для своїх дітей, тому коли вони виявлять, що веб-сайт не може донести до них важливу інформацію, вони можуть просто залишити його і шукати іншу школу.

Поряд з інформативною функцією, шкільний веб-сайт є найважливішим маркетинговим активом, але часто школи нехтують цією важливою складовою. Це цілком зрозуміло. Підтримка його актуальності може зайняти багато часу і витрат (незалежно від того, чи робите ви це самостійно, чи наймаєте для цього окремих спеціалістів).

Метою дипломного проєкту є створення веб-сайту Нижчелуб'янської ЗОШ І-ІІ ст. для сприяння залученню нових учнів та висвітлення цікавих новин і подій, що відбуваються в навчальному закладі. Завдяки розумно організованому SEO та вхідному маркетингу шкільний веб-сайт може стати потужним засобом для збільшення кількості відвідувачів.

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 7    |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |



# 1 АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ ТА ПОСТАНОВКА ЗАВДАННЯ

## 1.1 Обґрунтування доцільності створення web-сайту

Web-сайт школи — це звичайний інтернет-сайт, який спрямований на донесення важливої інформації про навчальний заклад в зручному вигляді для потенційних нових учнів, а також для вже своїх викладачів та школярів.

Незалежно від розміру та цілей школи, навчальний заклад повинен мати повноцінно функціонуючий веб-сайт, щоб продемонструвати основні дані про вступ, місцезнаходження та інформацію про персонал або забезпечити онлайн-уроки та курси.

Мета дипломного проєкту полягає у створенні сайту для потреб школи.

Виходячи з мети дипломного проєкту необхідно виконати наступні завдання:

- дослідити предметну область та здійснити постановку задачі;
- споектувати структуру та інтерфейс сайту;
- проаналізувати засоби реалізації завдання;
- реалізувати сайт;
- протестувати його.

## 1.2 Огляд існуючих рішень

Сьогодні майже кожна школа має свій інтернет-ресурс. Усі вони унікальні, як і інтерфейсом так і функціональністю та в свою чергу значно спрощують комунікацію між вчителями та учнями, особливо під час дистанційного навчання. Тому перед тим як починати розробляти власний сайт необхідно провести аналіз вже існуючих сайтів інших шкіл. Для прикладу розглянемо декілька веб-сайтів Тернопільських шкіл та ліцеїв:

- Тернопільська ЗОШ №11;
- Тернопільська ЗОШ №18;
- ТНВК «Школа-Ліцей №6».

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 8    |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

На рисунку 1.1 представлений вигляд головної сторінки веб-сайту одинадцятій школи.

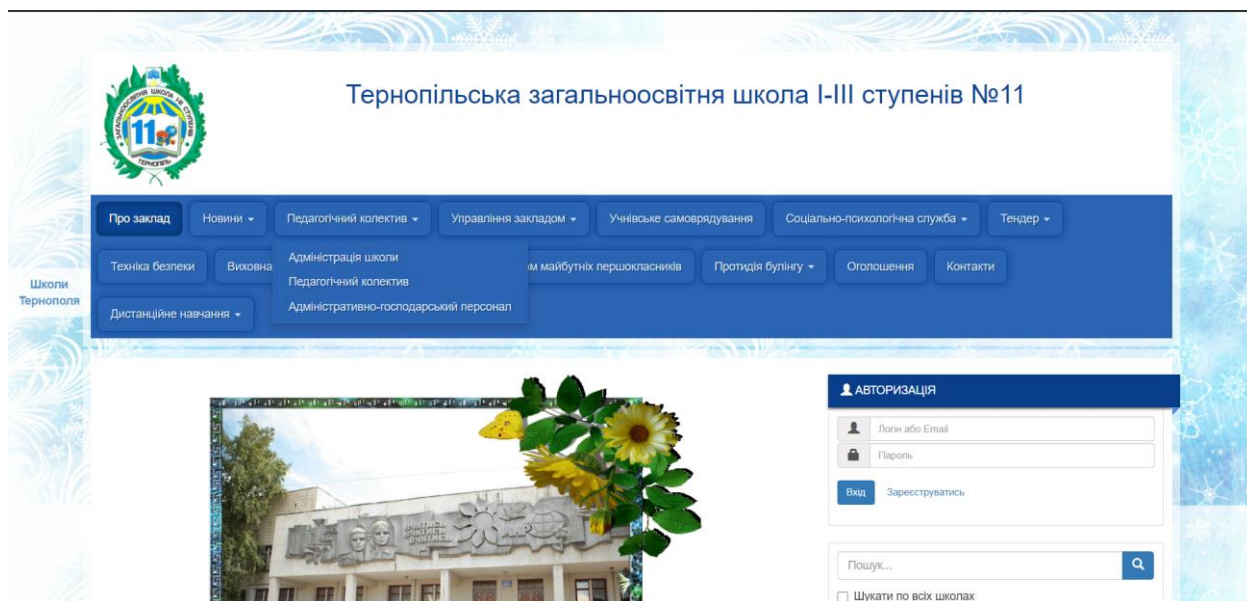


Рисунок 1.1 — Головна сторінка сайту Тернопільської ЗОШ №11

Сайт має інтуїтивний інтерфейс, на головній сторінці одразу міститься навігаційне меню з допомогою якого можна швидко знайти необхідну інформацію хто б це не був, вчитель, учень чи батьки.

Нижче міститься головний контент школи, фото та відео, історія та режим роботи школи (рисунок 1.2).

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 9    |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |



Рисунок 1.2 — Контент сайту Тернопільської ЗОШ №11

Актуальний на сьогодні пункт меню «Дистанційне навчання», учні можуть перейти у випадаючому списку на потрібний предмет та опрацювати його (рисунок 1.3).

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 10   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

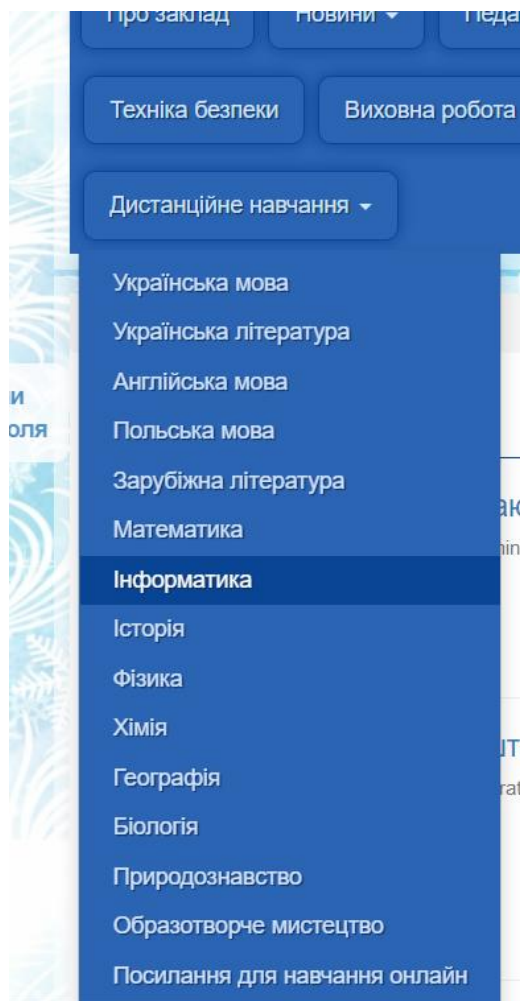


Рисунок 1.3 — Випадаючий список з предметами

Після переходу по предметі, завантажується сторінка вчителями та їх завданнями, зображення на рисунку 1.4.

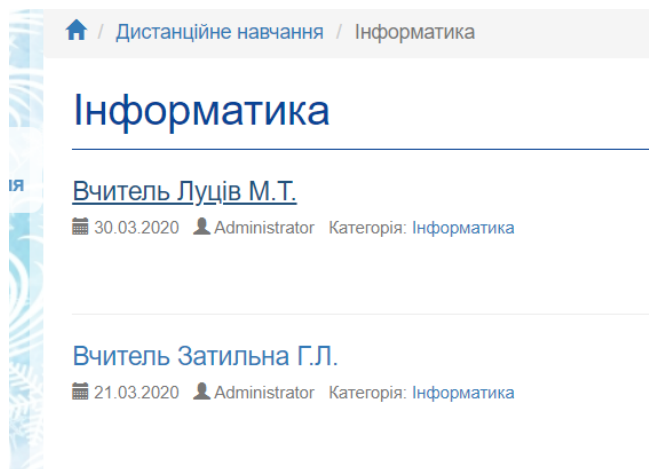


Рисунок 1.4 — Вчителі предмету

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 11   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

Також на сайті міститься форма реєстрації[5], а це говорить про те що на сайті є база даних користувачів (рисунок 1.5).

## Реєстрація

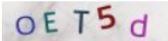
|   |  |
|---|--|
| Ваше ім'я   | <input type="text"/>   |
| Електронна пошта  | <input type="text"/>   |
| Логін   | <input type="text"/>   |
| Пароль  | <input type="password"/>   |
| Повторити пароль  | <input type="password"/>   |
| Введіть код з картинки  |  <input type="text"/> |
| <input type="button" value="Зареєструватись"/> <input type="button" value="Скасувати"/> |  |

Рисунок 1.5 — Форма реєстрації на сайті

Наступний веб-сайт, який ми розглянемо – сайт Тернопільської ЗОШ №18. Його головна сторінка зображена на рисунку 1.6.

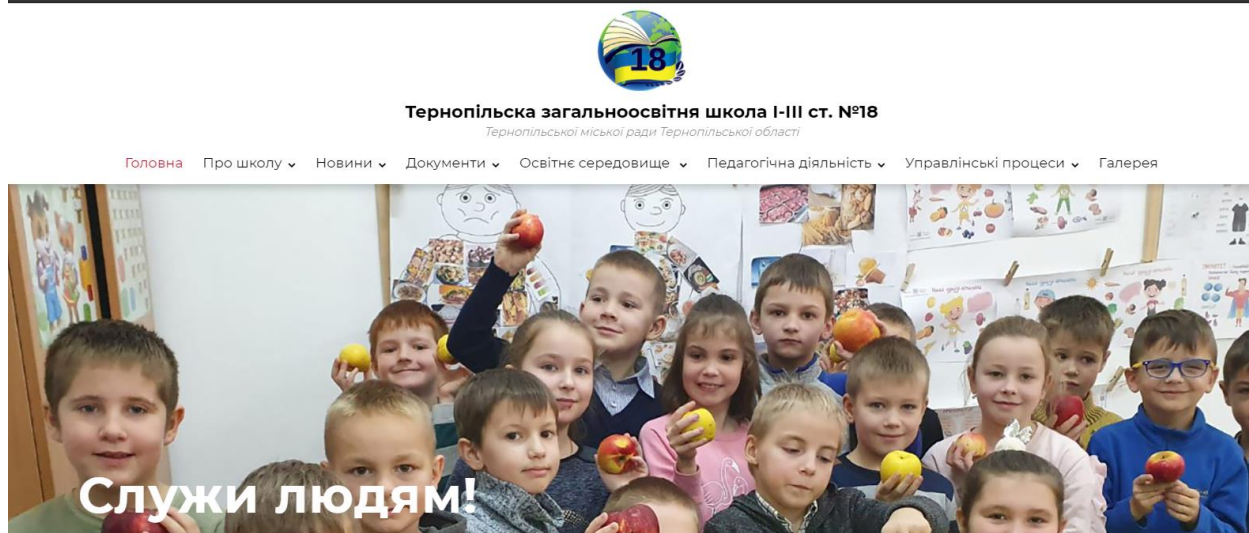


Рисунок 1.6 — Головна сторінка сайту Тернопільської ЗОШ №18.

Як бачимо, тут присутнє просте та зрозуміле меню з допомогою якого можна знайти головні новини школи (рисунок 1.7).

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 12   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |





**Хто вмирає в боротьбі – в серцях живе повіки!**



**Фруктовий день у ТЗОШ №18**

#### Останні новини

**Хто вмирає в боротьбі – в серцях живе повіки!**

лютий 18, 2021

**Фруктовий день у ТЗОШ №18**

лютий 03, 2021

**День безпеки в мережі Інтернет**

лютий 01, 2021

**Дистанційне навчання**

січень 11, 2021

Рисунок 1.7 — Останні новини школи

Також присутнє і освітнє середовище де знаходяться всі матеріали для навчання, зображення на рисунку 1.8.

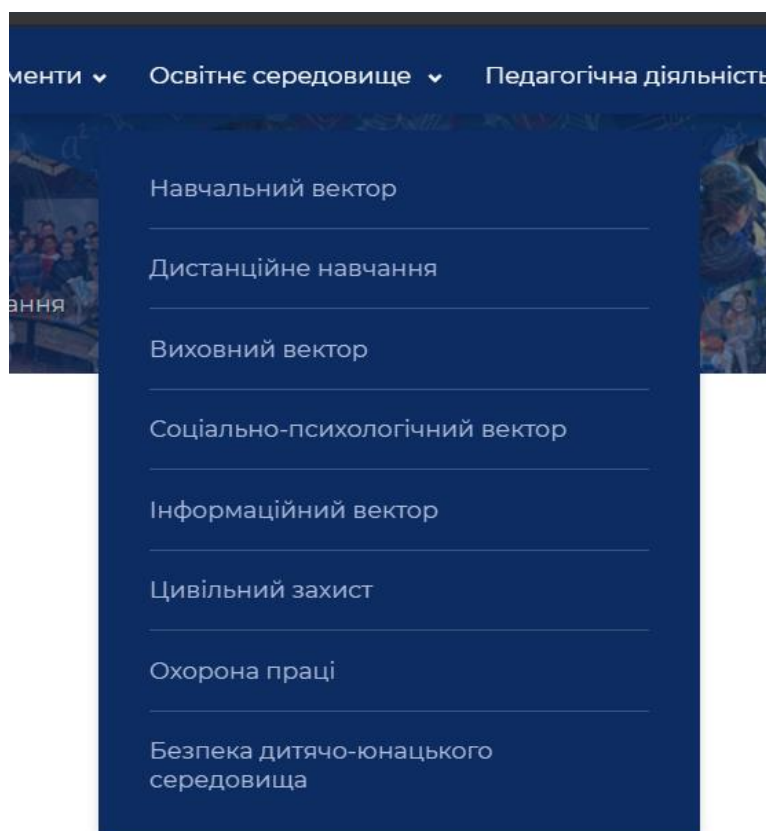


Рисунок 1.8 — Освітнє середовище школи на веб-сайті

Сайт теж достатньо інтуїтивний та функціональний, на ньому також міститься статистика кількості учнів та вчителів.

І останній сайт який розглянемо – вебсайт Школи-Ліцею №6 м. Тернополя. Головна сторінка зображена на рисунку 1.9.

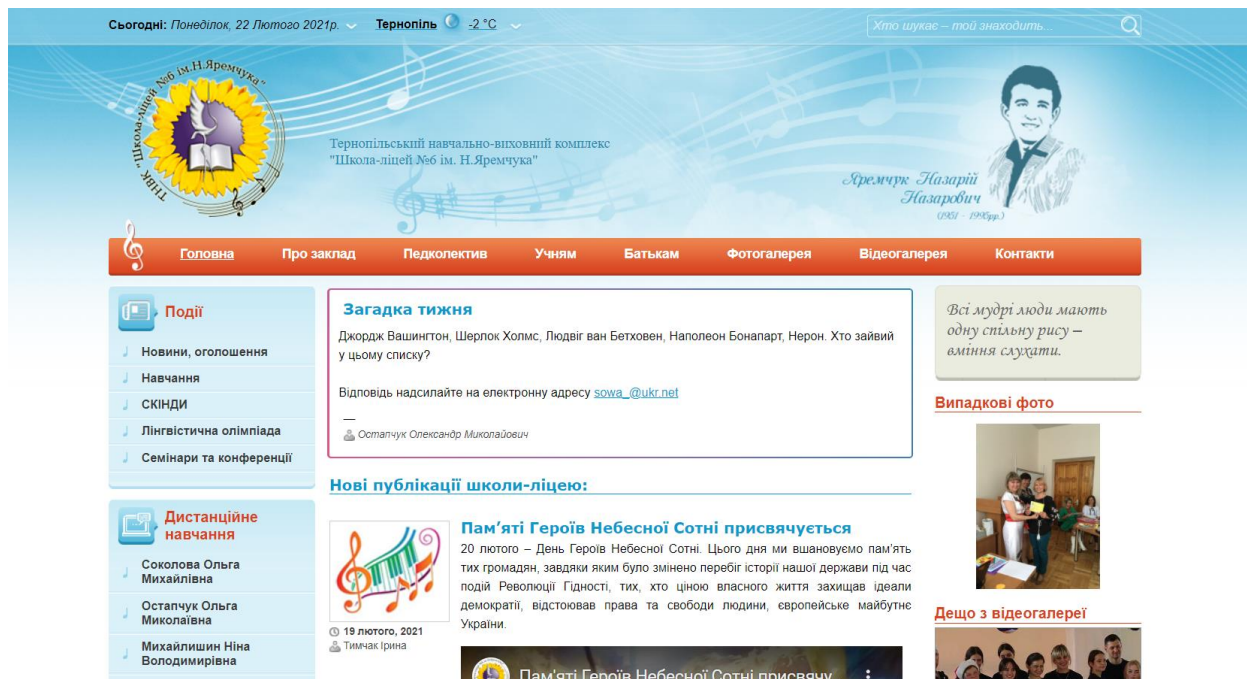
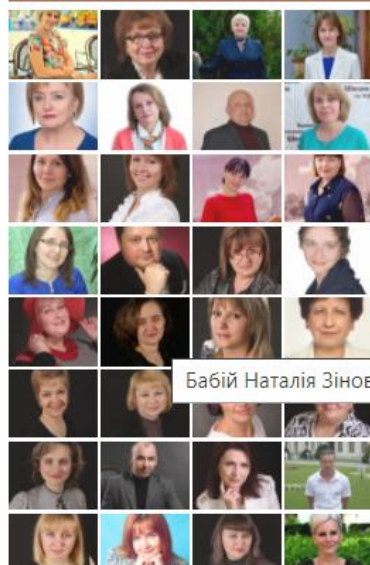


Рисунок 1.9 — Головна сторінка сайту Школи-Ліцею №6

Сайт є достатньо наповненим, містить багато блоків з різною інформацією (фото, відео, статті про діяльність школи). На головній [6] сторінці щотижня публікується загадка, відповідь на яку можна надіслати на вказану електронну адресу. Також є фото всіх вчителів в школі та їх посади (рисунок 1.10).

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        | 14   |

### Педагогічний колектив



Бабій Наталя Зіновіївна - вчитель початкових класів

Рисунок 1.10 — Педколектив Школи-Ліцею №6.

В самому верху сторінки знаходиться поле для пошуку по сайті та інформація про дату та погоду(рисунок 1.11).



Рисунок 1.11 — Хедер сайту

Про дистанційне навчання тут теж подумали та створили інформаційний блок із вчителями та їхніми завданнями (рисунок 1.12).

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 15   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |



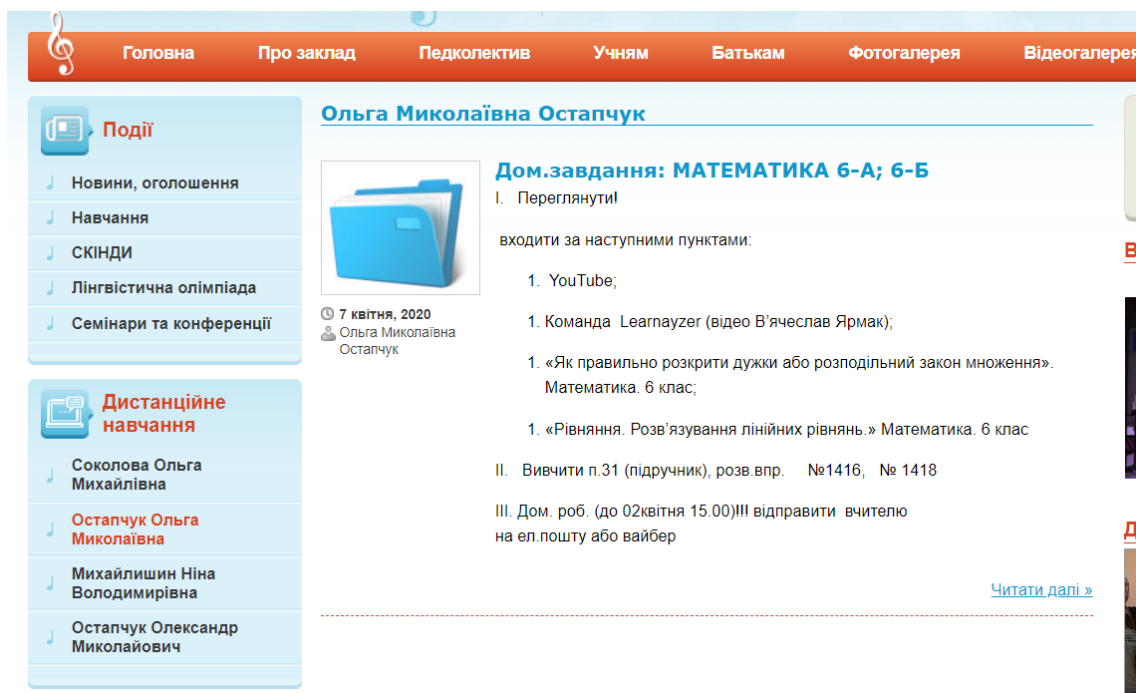


Рисунок 1.12 — Дистанційне навчання на сайті

Беручи до уваги описані готові рішення веб-сайтів для шкіл, в майбутньому веб-сайті потрібно звернути увагу на наступні моменти:

- сайт повинен мати зручний та інтуїтивний інтерфейс;
- можливість додавання контенту на сайт;
- зрозуміла структура навчання та діяльності в школі.

### 1.3 Постановка задачі

Завданням дипломного проєкту є розробка веб-сайту для школи. Цей інтернет-сайт призначений для швидкої та віддаленої комунікації вчителів з учнями та їх батьками. Публікації постів про життя школи та її діяльність, а також саме навчання. Для реалізації вищенаведених завдань заплановано провести аналіз різноманітних методів реалізації сайту та їх втілення у життя.

При декомпозиції завдання виділено основні процеси для розробки сайту, а саме можливість додавати контент на сайті та зручний його перегляд.

Сайт повинен виконувати наступні функції:

- перегляд основної інформації про заклад;
- перегляд головних новин закладу;

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 16   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

— зворотній зв'язок з закладом;

Функціональні можливості для адміністрації:

— додавання нової інформації;

— перегляд доданої інформації;

— видалення записів.

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 17   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

## 2 ПРОЄКТУВАННЯ WEB-САЙТУ

### 2.1 Проєктування структури web-сайту

Для даного сайту визначено наступну структуру:

- На односторінковому сайті зверху розміщено якірне меню, в якому розміщені всі частини сайту, воно дозволяє легко та швидко переміщуватися по ньому.
- При прокручуванні вниз сторінки відображуються останні новини школи які розміщені в слайдері.
- Нижче розміщений блок з інформацією для вступників та інформація про школу.

Діаграму варіантів використання (користувач, адміністратор) зображено на рисунках 2.1 та 2.2.

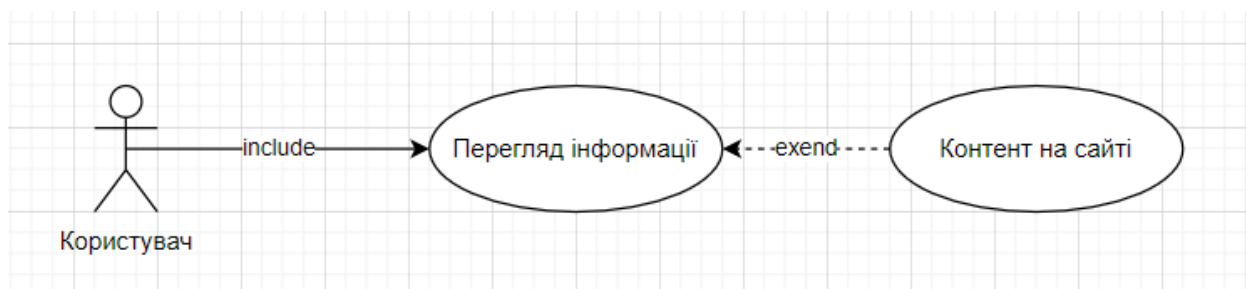


Рисунок 2.1 — Діаграма варіантів використання для користувача

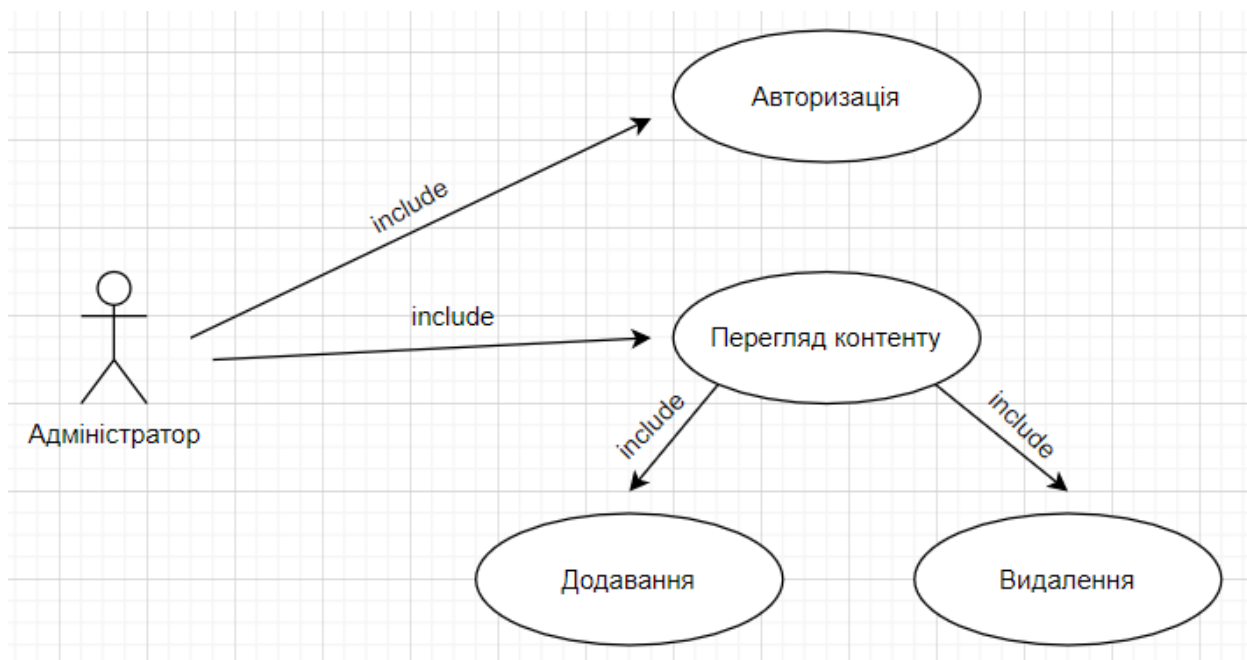


Рисунок 2.2 — Діаграма варіантів використання для адміністратора

Дана структура є досить простою та забезпечить виконання всіх функціональних вимог, поставлених до розроблюваного сайту.

## 2.2 Проєктування користувацького інтерфейсу

Веб-дизайн – це процес планування, концептуалізації та організації вмісту в Інтернеті. Сьогодні дизайн веб-сайту виходить за рамки естетики, включаючи загальну функціональність веб-сайту. Веб-дизайн також включає веб-програми, мобільні програми та дизайн інтерфейсу користувача.

Існує два основних способи створення веб-сайту: за допомогою настільного додатка (написання «вручну») або за допомогою конструктора веб-сайтів. Інструмент, який ви вирішите використовувати, буде сильно відрізнятися залежно від розміру команди, бюджету та типу сайту, який створюється, та його технічних вимог.

При розробці веб-сайту важливо враховувати як зовнішній вигляд, так і його функціональність. Інтеграція цих елементів дозволить максимізувати загальну зручність та продуктивність сайту. Юзабіліті веб-сайту включає такі елементи, як зручний для навігації інтерфейс, належне використання графіки та

зображень, добре написаний та добре розміщений текст та колірна гама. Ефективність сайту стосується його швидкості, рейтингу, доступності для пошуку та здатності охопити певну аудиторію.

Навігація на вебсайті – це один з основних елементів, який визначає, чи правильно працює сайт. Залежно від аудиторії, навігація може слугувати різним цілям: допомагати відвідувачам, які вперше дізнаються, що пропонує сайт, забезпечуючи легкий доступ до сторінок для відвідувачів, що повертаються, та покращуючи загальний досвід кожного відвідувача.

Нікому не подобається повільний веб-сайт. Якщо потрібно зачекати більше кількох секунд, поки сторінка завантажиться, це може перешкодити відвідувачеві залишитися на сайті та переглядати контент. Незалежно від того, наскільки хорошим є дизайн, якщо сайт завантажиться не швидко, він не буде добре працювати в пошуку.

Структура вебсайту відіграє важливу роль як у взаємодії з користувачем (UX), так і в оптимізації пошукової системи (SEO). Користувачі повинні мати можливість легко переміщатися по веб-сайту, не стикаючись з будь-якими структурними проблемами. Якщо користувачі губляться під час спроби перейти по сторінках сайту, швидше за все створено незручну навігацію, яка може призвести до поганого користувацького досвіду та рейтингу веб-додатку.

На рисунку 2.3 зображено структурну схему інтерфейсу користувача, спроектовану для Нижчелубянської ЗОШ І-ІІ ст.

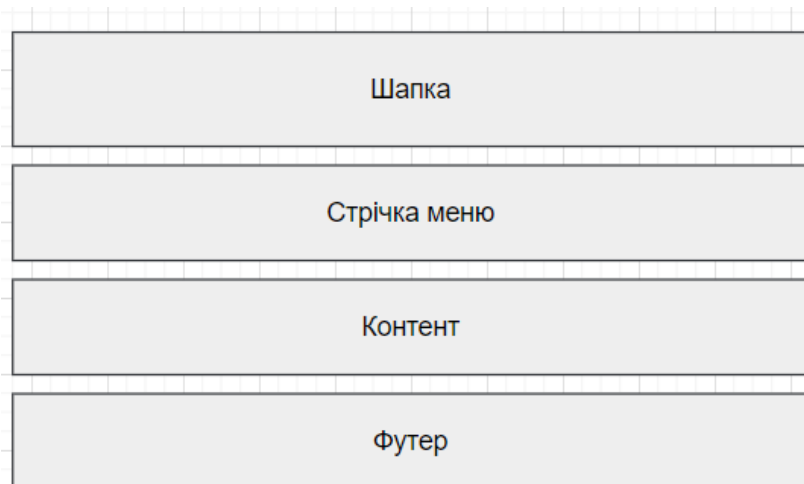


Рисунок 2.3 — Структурна схема інтерфейсу користувача сайту

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 20   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

Така структура є досить простою та інтуїтивно зрозумілою, користувач не буде «губитися» в елементах навігації та швидко зможе знайти необхідний контент.

### 2.3 Аналіз підходів до реалізації сайту

На сьогодні існує досить велика кількість методів розробки сайтів. Вибір методу розробки сайту в основному залежить від типу сайту, його запланованих функцій, дизайну та, звичайно, знань web-розробника.

Всі методи розробки сайтів можна умовно розділити на 2 основні групи. Перша група методів розробки сайтів – це методи ручного написання сайтів на одній або декількох мовах web-програмування. При цьому робота може здійснюватися як у простих (текстових), так і візуальних редакторах HTML та CSS. Останні дозволяють створювати сайти в режимі WYSIWYG – «Що Бачу Те й Одержую».

У випадку статичного сайту цілком достатнім для ручного написання буде використання «зв'язування» HTML і CSS, з можливим включенням JavaScript. Для створення ж динамічного сайту не обійтися без серверних скриптів, таких як PHP, ASP.NET і т.д.

Створювати й редагувати вручну файли .php можна навіть у звичайному «Блокноті», що поставляється з ОС Windows. Для роботи з ASP.NET прийде додатково встановити програмний продукт Microsoft Visual Studio, що встановлюється окремо.

Проаналізувавши існуючі методи, та інструментальні засоби, було вирішено реалізувати веб-сайт Нижчелуб'янської школи на платформі Node та з допомогою бібліотеки React.js з використанням мови гіпертекстової розмітки HTML, для оформлення використати CSS.

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 21   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

## 3 РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ

### 3.1 Опис засобів реалізації

React – це фреймворк з відкритим кодом JavaScript, який насправді не є фреймворком. Але на даний час це найбільш часто використовувана технологія розробки інтерфейсів у світі. React, спочатку розроблений і підтримуваний Facebook і підтримуваний активною спільнотою з відкритим кодом, насправді є «бібліотекою» JS.

ReactJS відповідає за побудову ієрархії або рендерингу компонентів інтерфейсу користувача та забезпечує підтримку як зовнішнього інтерфейсу, так і сторони сервера.

Як і всі фреймворки та бібліотеки JavaScript, основним варіантом використання React є створення динамічних користувацьких інтерфейсів, які оновлюються на стороні клієнта (користувача). Сторінку не потрібно оновлювати в браузері, вона автоматично оновлюється. Точно так, як це робить стрічка Facebook. Ось чому гігант соціальних мереж інвестував у розробку власної бібліотеки JavaScript, спеціально розробленої для потреб власних програм.

React зарекомендував себе як найбільш часто використовувана інтерфейсна платформа/бібліотека JS у світі, оскільки вона популярна як серед розробників програмного забезпечення, так і серед спонсорів проєктів.

Популярність React сьогодні затьмарила популярність усіх інших фронтенд-фреймворків. Ось чому:

- Простота створення динамічних додатків: React спрощує створення динамічних веб-додатків, оскільки вимагає менше кодування та пропонує більше функціональних можливостей, на відміну від JavaScript, де кодування часто дуже швидко ускладнюється.

- Покращена продуктивність: React використовує віртуальний DOM, тим самим швидше створюючи веб-програми. Віртуальний DOM порівнює попередні стани компонентів і оновлює лише ті елементи в Real DOM, які були

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 22   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

змінені, замість того, щоб знову оновлювати всі компоненти, як це роблять звичайні веб-програми.

– Багаторазові компоненти: Компоненти є будівельними блоками будь-якої програми React, і одна програма, як правило, складається з декількох компонентів. Ці компоненти мають свою логіку та елементи керування, і їх можна використовувати повторно протягом усієї програми, що, в свою чергу, різко скорочує час розробки програми.

– Односпрямований потік даних: React слідує за односпрямованим потоком даних. Це означає, що при розробці програми React розробники часто вкладають дочірні компоненти в батьківські компоненти. Оскільки дані рухаються в одному напрямку, стає простіше налагоджувати помилки та знати, де проблема виникає у програмі на даний момент.

– Швидке навчання: React легко засвоїти, оскільки він здебільшого поєднує основні концепції HTML та JavaScript з деякими корисними доповненнями. Проте, як і у випадку з іншими інструментами та фреймворками, вам доведеться витратити трохи часу, щоб отримати належне розуміння бібліотеки React.

– Його можна використовувати для розробки веб- і мобільних додатків: ми вже знаємо, що React використовується для розробки веб-додатків, але це не все, що він може зробити. Існує фреймворк під назвою React Native, похідний від самого React, який надзвичайно популярний і використовується для створення прекрасних мобільних додатків. Отже, насправді React можна використовувати для створення як веб-, так і мобільних додатків.

– Виділені інструменти для легкого налагодження: Facebook випустив розширення Chrome, яке можна використовувати для налагодження програм React. Це робить процес налагодження веб-програм React швидшим та простішим.

Вищезазначені причини з лишком виправдовують популярність бібліотеки React та те, чому вона використовується великою кількістю організацій та підприємств.

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 23   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |



JSX, або JavaScript XML, є розширенням синтаксису мови JavaScript. Подібно до зовнішнього вигляду HTML, JSX надає спосіб структурувати візуалізацію компонентів за допомогою синтаксису, знайомого багатьом розробникам. Компоненти React, як правило, пишуться за допомогою JSX, хоча їх не обов'язково (компоненти також можуть бути написані на чистому JavaScript). JSX подібний до іншого синтаксису розширення, створеного Facebook для PHP під назвою XHP.

Каскадні таблиці стилів (CSS) – це мова таблиць стилів, що використовується для опису презентації документа, написаного мовою розмітки, такою як HTML. CSS – це одна з технологій всесвітньої павутини, яка знаходиться поряд з HTML та JavaScript.

CSS призначений для розділення презентації та вмісту, включаючи макет, кольори та шрифти. Це розділення може покращити доступність вмісту, забезпечити більшу гнучкість та контроль у специфікації характеристик презентації, дозволити спільному форматуванню декілька веб-сторінок, вказавши відповідний CSS в окремому файлі .css, що зменшує складність та повторюваність структурного вмісту, а також дозволяє файл .css, який буде кешовано, щоб покращити швидкість завантаження сторінки між сторінками, що ділять файл, та його форматування.

Поділ форматування та вмісту також робить можливим подання однієї і тієї ж сторінки розмітки в різних стилях для різних методів візуалізації, таких як екран, друк, голосом (через мовний браузер або зчитувач екрана) та на основі Брайля тактильні пристосування. CSS також має правила альтернативного форматування, якщо доступ до вмісту здійснюється на мобільному пристрої.

Каскадне ім'я походить із зазначеної пріоритетної схеми, щоб визначити, яке правило стилю застосовується, якщо більше одного правила відповідає певному елементу. Ця схема каскадного пріоритету передбачувана.

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 24   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

Специфікації CSS підтримуються Консорціумом World Wide Web. На додаток до HTML, інші мови розмітки підтримують використання CSS, включаючи XHTML, звичайний XML, SVG та XUL.

Вперше CSS був запропонований Håkon Wium Lie 10 жовтня 1994 р. На той час Лі працювала з Тімом Бернерс-Лі в CERN . Приблизно в той же час було запропоновано кілька інших мов таблиць стилів для Інтернету, і обговорення публічних списків розсилки та всередині Консорціуму всесвітньої павутини призвели до того, що перша Рекомендація W3C CSS (CSS1) була опублікована в 1996 році. , пропозиція Берта Боса мала вплив; він став співавтором CSS1 і вважається співавтором CSS.

CSS має різні рівні та профілі. Кожен рівень CSS базується на останньому, як правило, додає нові функції і зазвичай позначається як CSS 1, CSS 2, CSS 3 і CSS 4. Профілі, як правило, є підмножиною одного або декількох рівнів CSS, побудованих для певного пристрою або користувацького інтерфейсу. На даний час існують профілі для мобільних пристроїв, принтерів та телевізорів. Не слід плутати профілі з типами носіїв, які були додані в CSS 2.

Кожен веб-браузер використовує механізм макетування для візуалізації веб-сторінок, і підтримка функціональності CSS не узгоджується між ними. Оскільки браузери не аналізують CSS ідеально, було розроблено декілька методів кодування для орієнтування на певні браузери з обхідними шляхами (загальновідомими як хакі CSS або фільтри CSS). Впровадженню нових функціональних можливостей у CSS може заважати відсутність підтримки у основних браузерах. Наприклад, Internet Explorer повільно додавав підтримку багатьох функцій CSS 3, що уповільнювало використання цих функцій та погіршувало репутацію браузера серед розробників. Щоб забезпечити незмінний досвід для своїх користувачів, веб-розробники часто тестують свої сайти в декількох операційних системах, браузерах та версіях браузерів, збільшуючи час розробки та складність. Такі інструменти, як BrowserStack , були створені для зменшення складності обслуговування цих середовищ.

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 25   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

Оскільки веб-сайти приймають нові стандарти коду, несумісні зі старими браузерами, ці браузери можуть бути відключені від доступу до багатьох ресурсів в Інтернеті (іноді навмисно). Багато найпопулярніших веб-сайтів в Інтернеті не просто візуально погіршуються у старих браузерах через погану підтримку CSS, але взагалі не працюють, значною мірою через еволюцію JavaScript та інших веб-технологій.

Express – найпопулярніший веб-фреймворк для Node. Він є базовою бібліотекою для ряду інших популярних веб-фреймворків Node. Він надає наступні механізми:

Написання обробників для запитів з різними HTTP-методами в різних URL-адреси (маршрутах).

Інтеграцію з механізмами рендеринга «view», для генерації відповідей, вставляючи дані в шаблони.

Встановлення загальних параметрів веб-додатку, такі як порт для підключення, і розташування шаблонів, які використовуються для відображення відповіді.

«Проміжне ПО» для додаткової обробки запиту в будь-який момент в конвеєрі обробки запитів.

У той час як сам Express досить мінімалістичний, розробники створили сумісні пакети проміжного програмного забезпечення для вирішення практично будь-якої проблеми з веб-розробкою. Існують бібліотеки для роботи з куکی-файлами, сеансами, входами користувачів, параметрами URL, даними POST, заголовками безпеки і багатьма іншими. Ви можете знайти список пакетів проміжного програмного забезпечення, підтримуваних командою Express в Express Middleware (поряд зі списком деяких популярних пакетів сторонніх виробників).

У традиційних динамічних веб-сайтах, веб-додаток очікує HTTP-запиту від веб-браузера (або іншого клієнта). Коли запит отримано, додаток визначає, яку дію необхідно виконати на основі URL шаблону і, можливо, пов'язаної

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 26   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

інформації, що міститься в даних POST або GET. Залежно від того, що потрібно, Express може потім читати або записувати дані з / в бази даних або виконувати інші завдання, відповідно до отриманого запиту. Потім додаток повертає відповідь в веб-браузер, найчастіше динамічно створюючи HTML сторінку для відображення браузером, вставляючи витягнуті дані в наповнювачі HTML шаблону.

Express надає методи дозволяють вказати, яка функція викликається для конкретного HTTP запиту (GET, POST, SET, etc.), і URL шаблон ("Route"), а також методи дозволяють вказати, який механізм шаблону ("view") використовується, де знаходяться шаблони файлів і який шаблон використовувати для виведення відповіді. Ви можете використовувати Express middleware для додавання підтримки файлів cookies, сеансів, і користувачів, отримання POST / GET параметрів, і т.д. Ви можете використовувати

будь-який механізм бази даних, підтримуваний Node (Express не визначає поведінку, пов'язану з базою даних).

MongoDB - це орієнтована на документи база даних NoSQL, яка використовується для великого обсягу зберігання даних. Замість використання таблиць и рядків, як у традиційних реляційних базах даних, MongoDB використовує колекції та документи. Документи складаються з пар ключ-значення, які є основною одиницею даних у MongoDB. Колекції містять набори документів та функції, що є еквівалентом реляційних таблиць баз даних. MongoDB - це база даних, яка з'явилася приблизно в середині 2000-х.

Особливості MongoDB [4]:

- Кожна база даних містить колекції, які в свою чергу містять документи. Кожен документ може бути різним із різною кількістю полів. Розмір і зміст кожного документа можуть відрізнятися один від одного.

- Структура документа більше відповідає тому, як розробники будують свої класи та об'єкти у відповідних мовах програмування. Розробники часто говорять, що їх класи не є рядками та стовпцями, а мають чітку структуру з парами ключ-значення.

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 27   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

- Рядки (або документи, як їх називають у MongoDB[3]), не повинні мати попередньо визначену схему. Натомість поля можна створювати на льоту.
- Модель даних, доступна в MongoDB, дозволяє вам легше представляти ієрархічні відносини, зберігати масиви та інші більш складні структури.
- На відміну від баз даних SQL, де перед вставкою даних вам потрібно оголосити схему таблиці, колекції MongoDB не застосовують структуру документу.

### 3.2 Створення проекту за допомогою платформи Node JS

Node (або більш формально Node.js ) – це середовище виконання з відкритим вихідним кодом, яке дозволяє розробникам створювати всі види серверних інструментів та додатків у JavaScript . Час виконання призначений для використання поза контекстом браузера (тобто працює безпосередньо на комп'ютері або серверній ОС). Таким чином, середовище опускає API-інтерфейси JavaScript для певного браузера та додає підтримку більш традиційних API-інтерфейсів ОС, включаючи HTTP[8] та бібліотеки файлової системи.

Перед початком написання коду потрібно встановити Node з офіційного сайту, він доступний як для Windows так і macOS. Сторінка завантаження на офіційному сайті зображена на рисунку 3.1

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 28   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

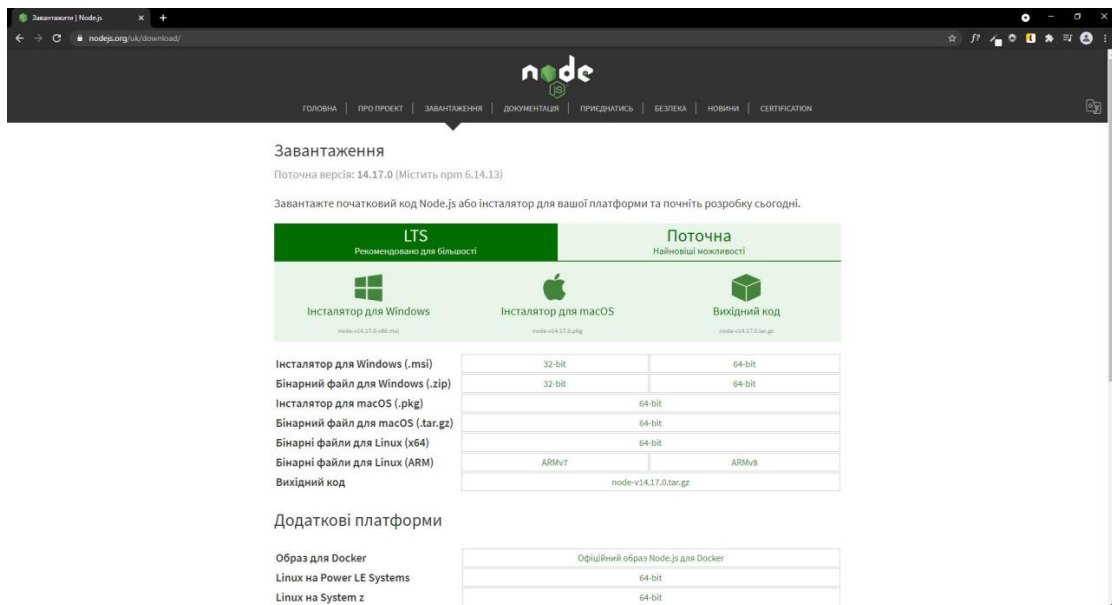


Рисунок 3.1 — Сторінка завантаження Node JS

Погоджуємось з умовами – рисунок 3.2 та вибираємо шлях встановлення – рисунок 3.3 (рекомендовано вказувати шлях латиницею).

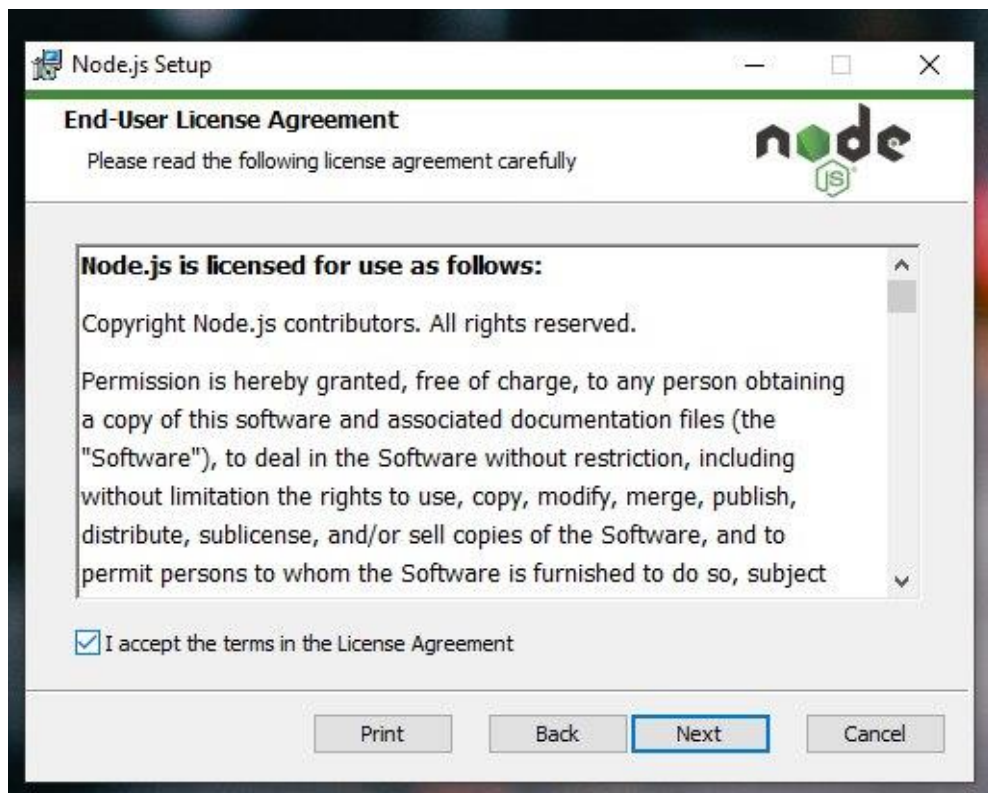


Рисунок 3.2 — Погодження з ліцензійними умовами

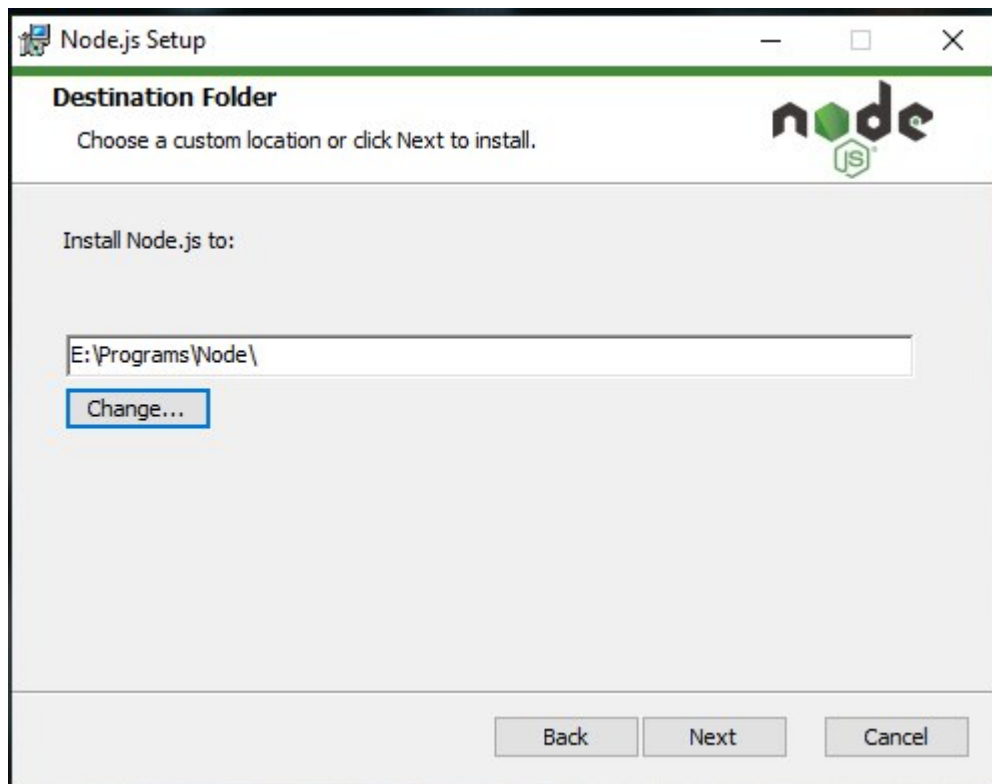


Рисунок 3.3 — Вибір шляху встановлення

Потім вибираємо всі рекомендовані налаштування та встановлюємо —  
 рисунок 3.4

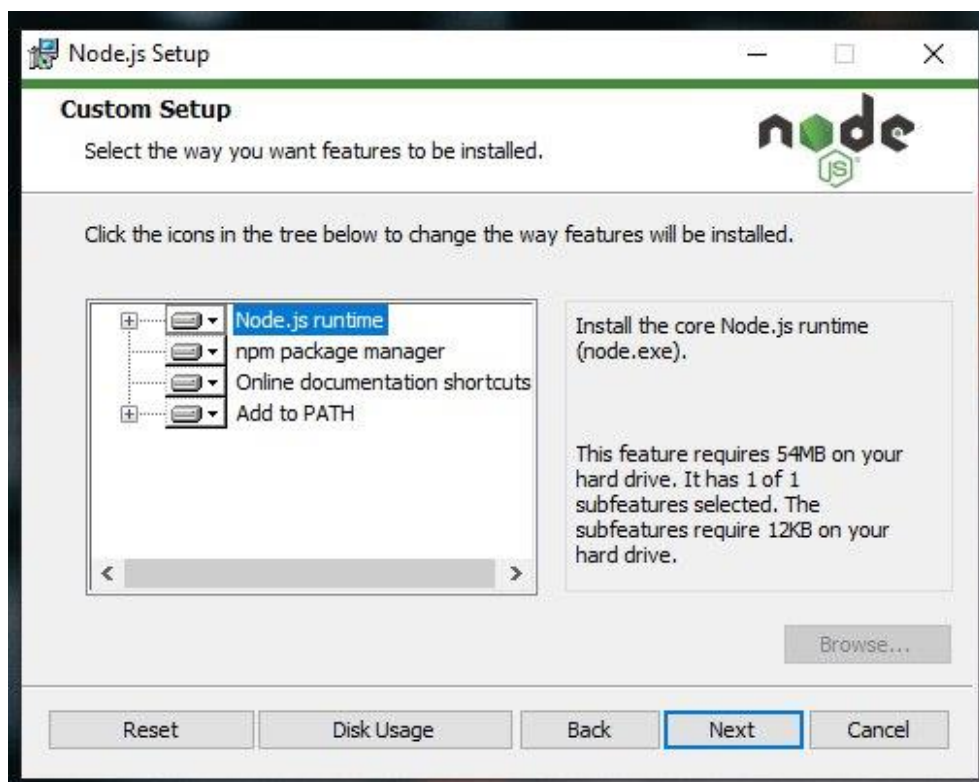


Рисунок 3.4 — Вибір налаштувань Node

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 30   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

Процес встановлення зображено на рисунку 3.5.

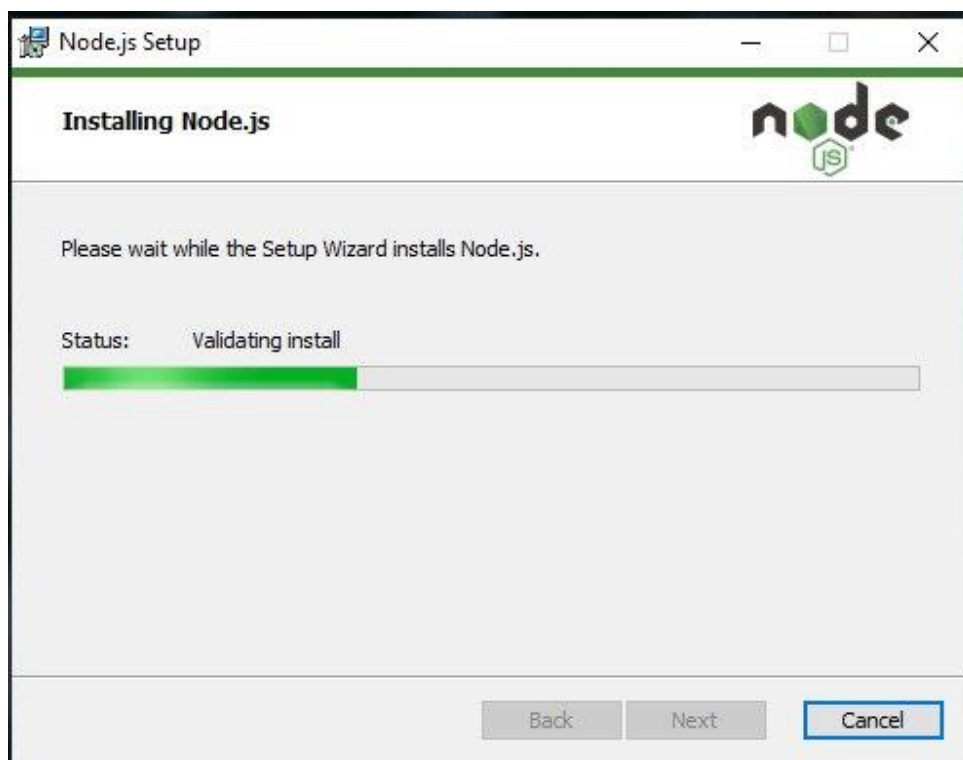


Рисунок 3.5 — Процес встановлення Node JS

Після встановлення Node JS[7] потрібно створити в консолі проект з бібліотекою React.

Створення проекту зображено на рисунку 3.6

```
E:\>cd project

E:\project>npx create-react-app school
[#####.....] / refresh-package-json:readable-stream: sill refresh-package-json C:\Users\38063.
DE[#####.....] / refresh-package-json:readable-stream: sill refresh-package-json C:\User[#####
[#####.....] / refr[#####.....] / refresh-package-json:readable-stream: sill refresh-p[#####np
x: installed 67 in 31.488s

Creating a new React app in E:\project\school.

Installing packages. This might take a couple of minutes.
Installing react, react-dom, and react-scripts with cra-template...

[#####.....] | fetchMetadata: sill resolveWithNewModule punycode@2.1.1 checking installable st
[#####.....] | fetchMetadata: sill resolveWithNewModule punycode@2.1.1 checking installable st
[#####.....] | fetchMetadata: sill resolveWithNewModule @babel/helper-plugin-utils@7.13.0 che
```

Рисунок 3.6 — Створення проекту React

Після попередніх дій створюється проект в якому вже можна починати робити вебсайт (рисунок 3.7).



```

darwin ; arch : any } (current: { OS : win32 ; arch : x64 })
removed 1 package and audited 1942 packages in 2060.094s

139 packages are looking for funding
  run `npm fund` for details

found 81 moderate severity vulnerabilities
  run `npm audit fix` to fix them, or `npm audit` for details

Success! Created school at E:\project\school
Inside that directory, you can run several commands:

  npm start
    Starts the development server.

  npm run build
    Bundles the app into static files for production.

  npm test
    Starts the test runner.

  npm run eject
    Removes this tool and copies build dependencies, configuration files
    and scripts into the app directory. If you do this, you can't go back!

We suggest that you begin by typing:

  cd school
  npm start

Happy hacking!
E:\project>_

```

Рисунок 3.7 — Створений проект React

### 3.3 Опис та реалізація власного дизайну для сайту

Для початку встановимо пакет Mongoose для серверної частини  
 рисунок 3.8 та 3.9

```

E:\project>cd school

E:\project\school>npm install mongoose express cors axios -S
[.....] / rollbackFailedOptional: verb npm-session 908d41f161919748

```

Рисунок 3.8 — Встановлення пакету Mongoose

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 32   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

```

E:\project\school>npm install mongoose express cors axios -S
npm WARN @babel/plugin-bugfix-v8-spread-parameters-in-optional-chaining@7.13.12 requires a peer of @babel/core@^7.13.0 but none is installed. You must install peer dependencies yourself.
npm WARN tsutils@3.21.0 requires a peer of typescript@>=2.8.0 || >= 3.2.0-dev || >= 3.3.0-dev || >= 3.4.0-dev || >= 3.5.0-dev || >= 3.6.0-dev || >= 3.6.0-beta || >= 3.7.0-dev || >= 3.7.0-beta but none is installed. You must install peer dependencies yourself.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@2.3.2 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.3.2: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.13 (node_modules\watchpack-chokidar2\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.13: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.13 (node_modules\webpack-dev-server\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.13: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})

+ cors@2.8.5
+ axios@0.21.1
+ mongoose@5.12.12
+ express@4.17.1
added 26 packages from 53 contributors, updated 1 package and audited 1969 packages in 64.339s

140 packages are looking for funding
  run `npm fund` for details

found 81 moderate severity vulnerabilities
  run `npm audit fix` to fix them, or `npm audit` for details

```

Рисунок 3.9 — Встановлення пакету Mongoose

Mongoose представляє спеціальну ODM-бібліотеку (Object Data Modelling) для роботи з MongoDB, яка дозволяє зіставляти об'єкти класів і документи колекцій з бази даних. Грубо кажучи, Mongoose працює подібно інструментам ORM [2].

Каркасом для розробки серверної частини буде Express.js.

Express являє собою популярний веб-фреймворк, написаний на JavaScript і працює всередині середовища node.js. Цей модуль висвітлює деякі ключові переваги цього фреймворка, установку середовища розробки і виконання основних завдань веб-розробки.

Фронт-енд частина розроблена на HTML5 та CSS3, лістинг коду подано в додатку А, зображення з головної сторінки сайту на рисунку 3.10 та 3.11

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 33   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

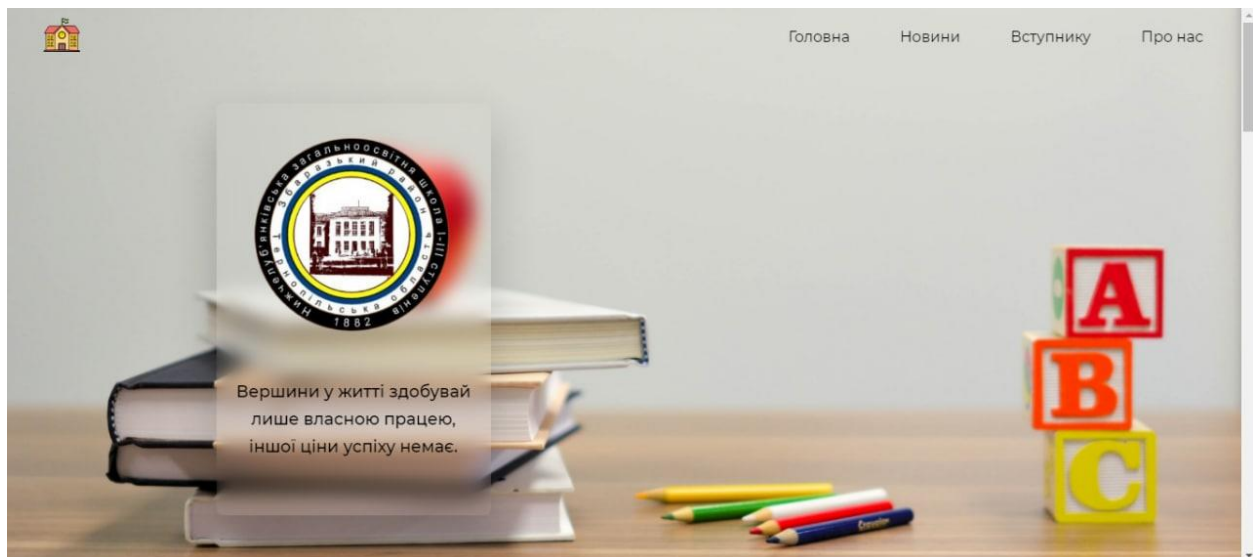


Рисунок 3.10 — Знімок із головної сторінки сайту

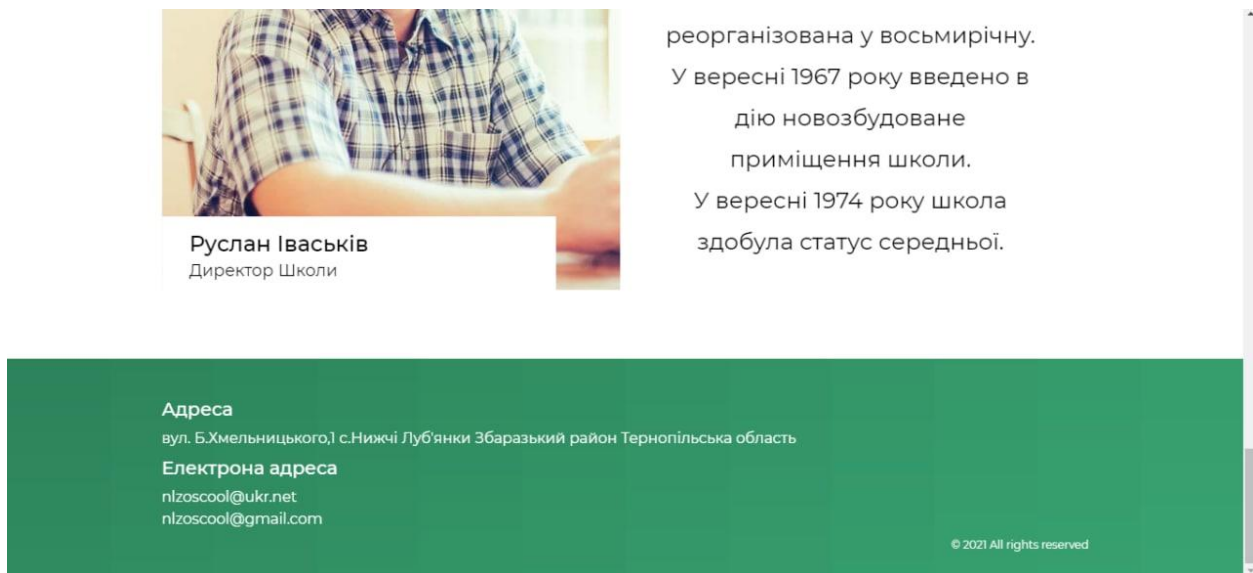


Рисунок 3.11 — Знімок із головної сторінки сайту

На сайті також розміщений слайдер з новинами школи. Встановлення його бібліотеки зображено на рисунку 3.12

```
E:\project\school>npm install react-slick --save
npm WARN @babel/plugin-bugfix-v8-spread-parameters-in-optional-chaining@7.13.12 requires a peer of @babel/core@^7.13.0 but none is installed. You must install peer dependencies yourself.
npm WARN tsutils@3.21.0 requires a peer of typescript@>=2.8.0 || >= 3.2.0-dev || >= 3.3.0-dev || >= 3.4.0-dev || >= 3.5.0-dev || >= 3.6.0-dev || >= 3.6.0-beta || >= 3.7.0-dev || >= 3.7.0-beta but none is installed. You must install peer dependencies yourself.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@2.3.2 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.3.2: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.13 (node_modules\watchpack-chokidar2\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.13: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.13 (node_modules\webpack-dev-server\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.13: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})

+ react-slick@0.28.1
added 6 packages from 4 contributors and audited 1974 packages in 26.081s

140 packages are looking for funding
  run `npm fund` for details

found 81 moderate severity vulnerabilities
  run `npm audit fix` to fix them, or `npm audit` for details
```

Рисунок 3.12 — Встановлення бібліотеки зі слайдером

### 3.4 Опис додавання контенту для сайту

В адмінпанелі написана форма для завантаження постів та новин на сайт  
 рисунок 3.13

Виберіть зображення

Рисунок 3.13 — Форма для додавання контенту

Можна вибрати медіафайл у форматі jpg та jpeg і додати заголовок та вміст статті. Після натиснення на кнопку “Зберегти” з’являється відповідне повідомлення (рисунок 3.14) і стаття публікується на сайті.

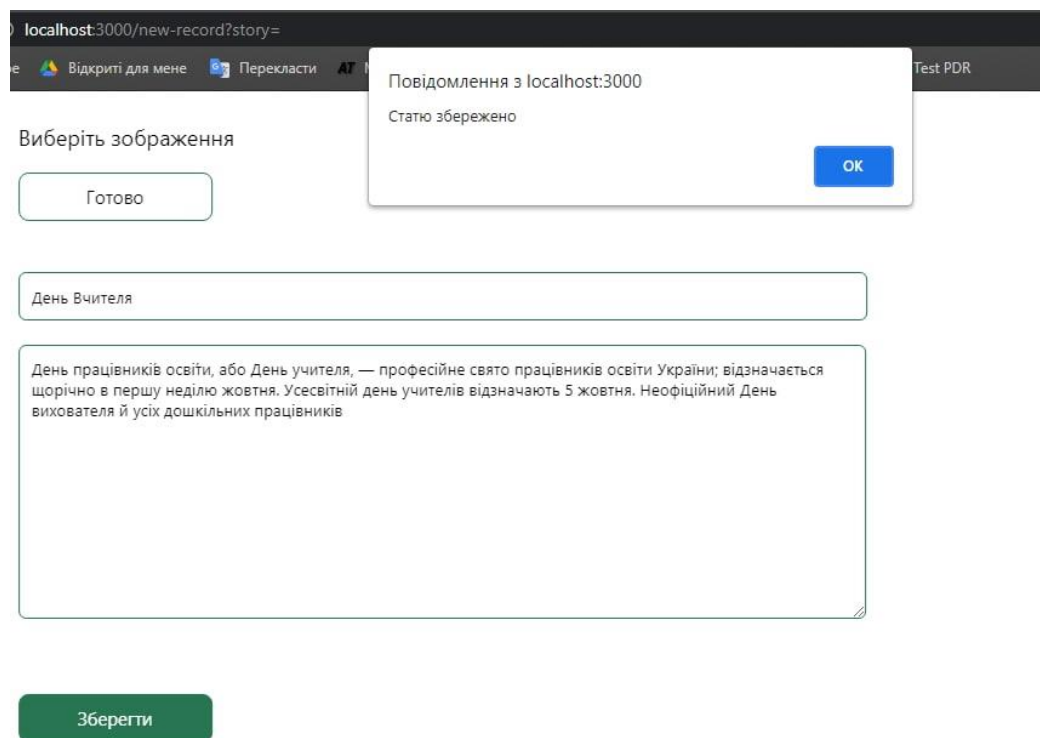


Рисунок 3.14 — Повідомлення про додавання статті

На рисунку 3.15 та 3.16 зображено як додана стаття буде виглядати безпосередньо на сайті.

## Новини

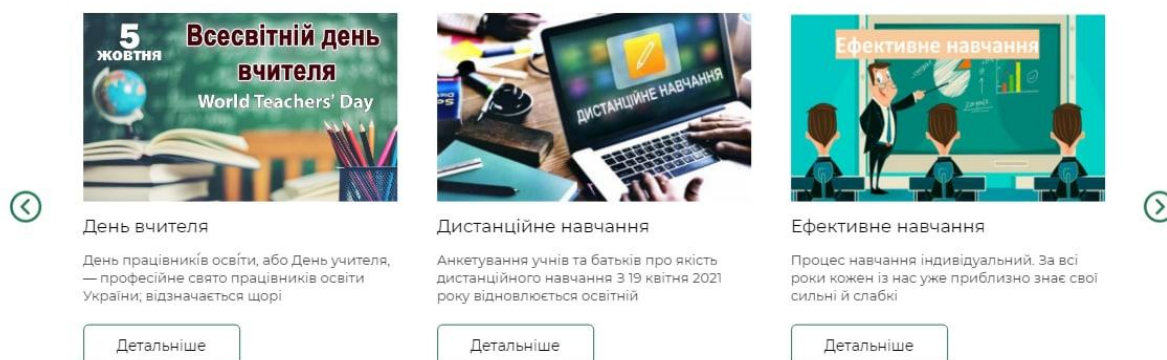


Рисунок 3.15 — Додана стаття

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 36   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |



## Новини

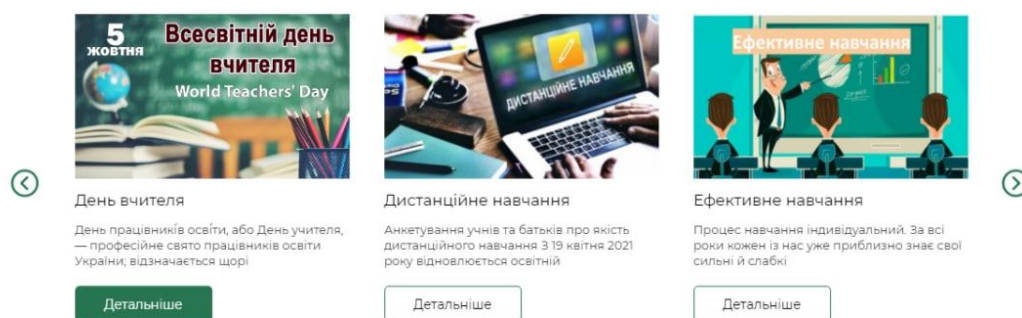


Рисунок 3.16 — Активна додана стаття

При натисканні на кнопку «Детальніше» потрапляєте на сторінку статті. Вона зображена на рисунку 3.17 та 3.18.



Рисунок 3.17 — Сторінка статті

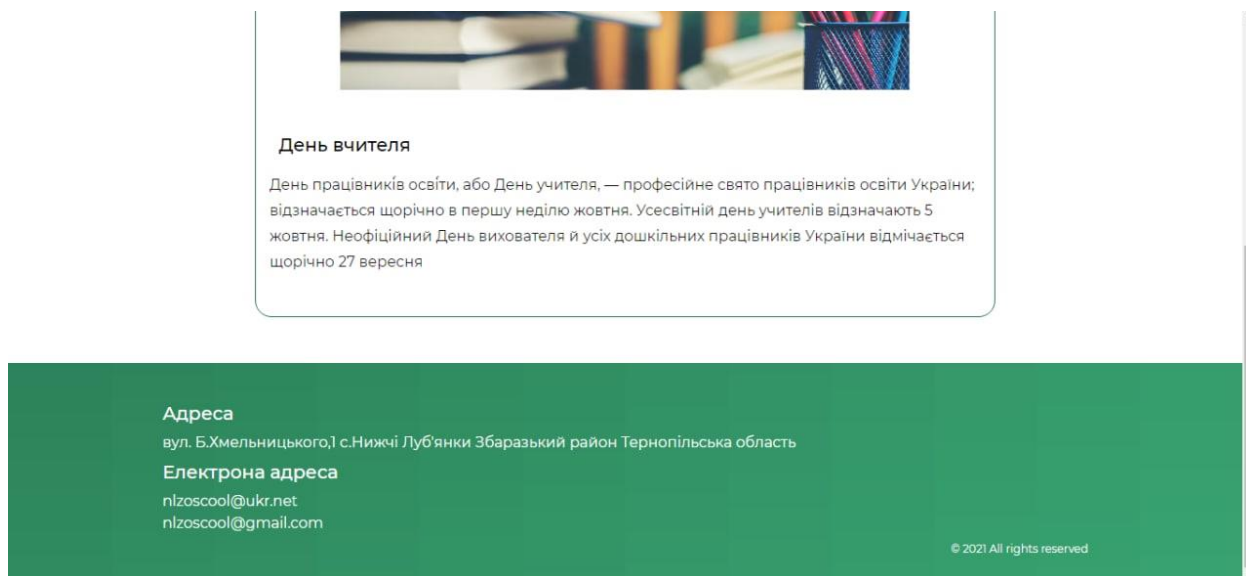


Рисунок 3.18 — Сторінка статті

Щоб перейти в адмін-панель сайту потрібно в стрічці адреси прописати адмін рисунок 3.19.

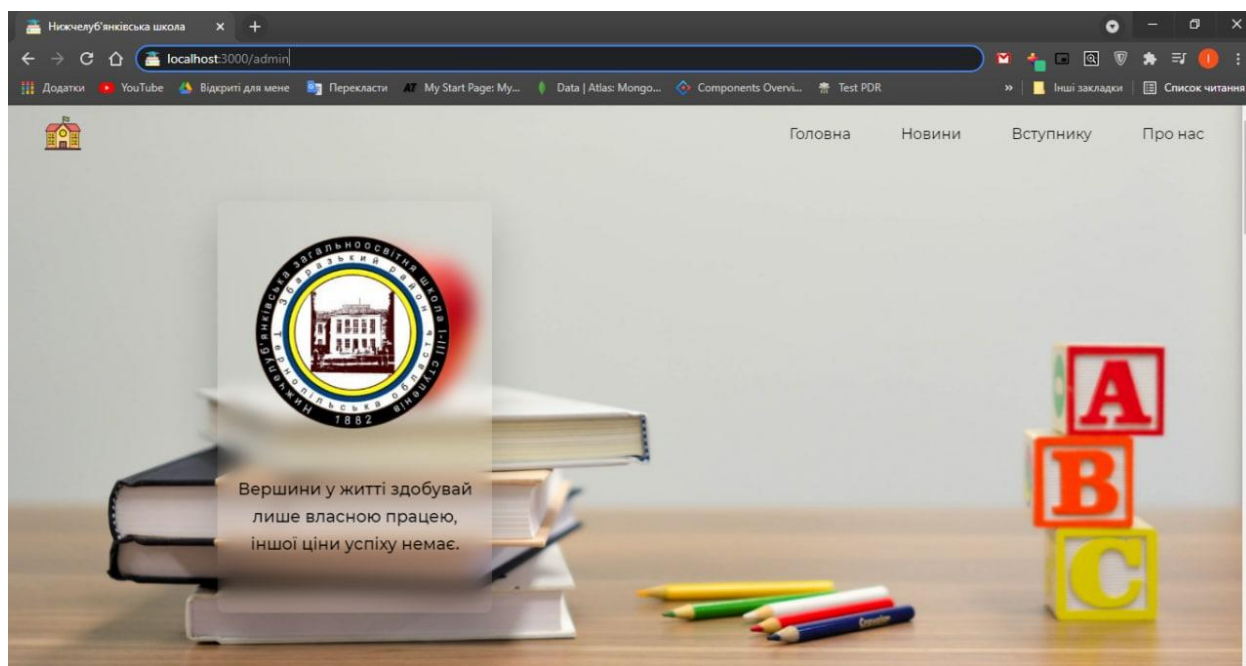


Рисунок 3.19 — Перехід в адмін-панель

Після цього переходите на сторінку адмін-панелі рисунок 3.20.

| Назва          | Дата       |
|----------------|------------|
| Скоро канікули | 2021-06-08 |
| День вчителя   | 2021-06-08 |
| День Вишиванки | 2021-06-08 |

Рисунок 3.19 — Адмін-панель

У разі якщо виникла якась помилка при зчитуванні з бази даних викидає помилку зображену на рисунку 3.20.

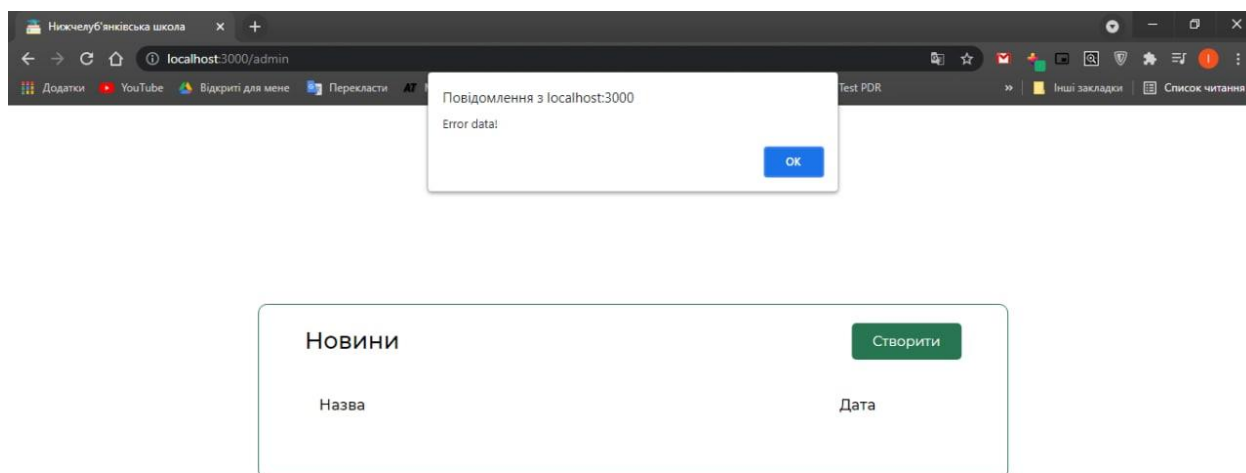


Рисунок 3.20 — Помилка зчитування з БД



Щоб запустити сервер вводимо команди в термінал зображені на рисунку 3.21.

```
C:\Windows\system32\cmd.exe - node server.js
F:\project\school>
F:\project\school>
F:\project\school>
F:\project\school>node server.js
Server is starting at 8080 port
Mongoose is conected!!
```

Рисунок 3.21 — Запуск сервера

Щоб не перезапускати сервер кожного разу після зміни чи додаванні нового коду було встановлено пакет `nodemon`, з ним дані оновлюються в реальному часі. Його процес встановлення зображено на рисунку 3.22-3.24.

```
E:\project>cd school
E:\project\school>npm install -g nodemon
[.....] | extract:nodemon: verb lock using C:\Users\38063.DESKTOP-K20TU6S\AppData\Roaming\npm-cache\_locks
```

Рисунок 3.22 — Встановлення podemon

```
Love nodeemon? You can now support the project via the open collective:
> https://opencollective.com/nodeemon/donate

npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@~2.3.1 (node_modules\nodeemon\node_modules\chokidar\node_modules
\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.3.2: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})

+ nodeemon@2.0.7
added 119 packages from 53 contributors in 35.117s
```

Рисунок 3.23 — Встановлення podemon

```
F:\project\school>nodemon server.js
[nodemon] 2.0.7
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node server.js`
Server is starting at 8080 port
Mongoose is conected!!
```

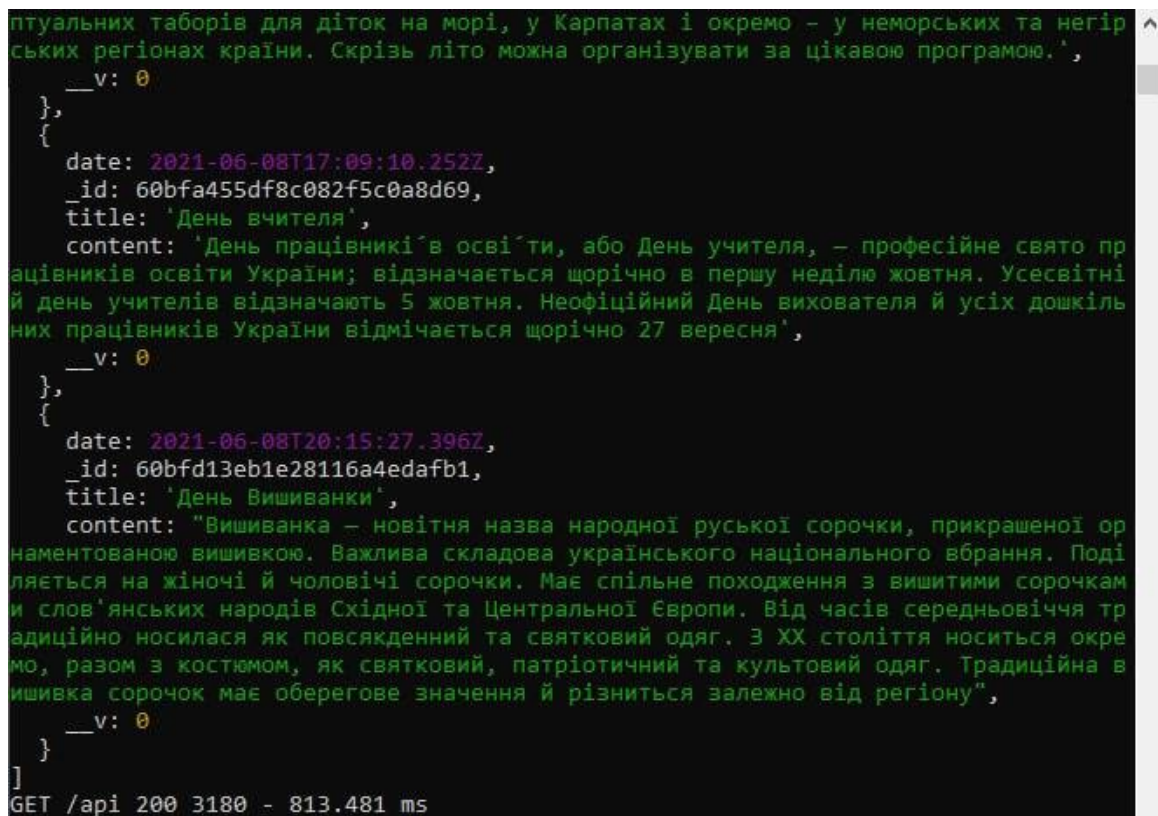
Рисунок 3.24 — Встановлений пакет

Для отримання даних з бази даних використовується код наведений в лістингу 3.1

### Лістинг 3.1 – Отримання даних з БД.

```
router.get('/', (req, res)=>{
  New.find({ })
    .then((data)=>{
      console.log('Data', data);
      res.json(data);
    })
    .catch((error)=>{
      console.log('error:', daerrorta);
    });
});
```

Після цього сервер отримає дані з БД(рисунок 3.25).



```
[{"_v": 0}, {"date": "2021-06-08T17:09:10.252Z", "_id": "60bfa455df8c082f5c0a8d69", "title": "\u0414\u0435\u043d\u044c \u0432\u0447\u0438\u0442\u0435\u043b\u044f", "content": "\u0414\u0435\u043d\u044c \u043f\u0440\u0430\u0446\u0456\u0432\u043d\u0438\u043a\u0456\u0432 \u0432 \u043e\u0441\u0432\u0456\u0442\u0456, \u0430\u0431\u043e \u0414\u0435\u043d\u044c \u0443\u0447\u0438\u0442\u0435\u043b\u044f, \u2013 \u043f\u0440\u043e\u0444\u0435\u0441\u0456\u0439\u043d\u0435 \u0441\u0432\u044f\u0442\u043e \u043f\u0440\u0430\u0446\u0456\u0432\u043d\u0438\u043a\u0456\u0432 \u043e\u0441\u0432\u0456\u0442\u0456 \u0423\u043a\u0440\u0430\u0456\u043d\u0438; \u0432\u0456\u0434\u043d\u0430\u0447\u0430\u0454\u0442\u044c\u0441\u044f \u0449\u043e\u0440\u0456\u0447\u043d\u043e \u0432 \u043f\u0435\u0440\u0448\u0443 \u043d\u0435\u0434\u0456\u043b\u044e \u0436\u043e\u0432\u0442\u043d\u044f. \u0423\u0441\u0435\u0441\u0432\u0456\u0442\u043d\u0456 \u0456 \u0434\u0435\u043d\u044c \u0443\u0447\u0438\u0442\u0435\u043b\u0456\u0432 \u0432\u0456\u0434\u043d\u0430\u0447\u0430\u0454\u0442\u044c\u0441\u044f 5 \u0436\u043e\u0432\u0442\u043d\u044f. \u041d\u0435\u043e\u0444\u0456\u0446\u0456\u0439\u043d\u0438\u0439 \u0414\u0435\u043d\u044c \u0432\u0438\u0432\u043e\u0432\u0430\u0442\u0435\u043b\u044f \u0456 \u0443\u0441\u0456\u0445 \u0434\u043e\u0448\u043a\u0456\u043b\u044c\u043d\u0438\u0445 \u043f\u0440\u0430\u0446\u0456\u0432\u043d\u0438\u043a\u0456\u0432 \u0423\u043a\u0440\u0430\u0456\u043d\u0438 \u0432\u0456\u0434\u043c\u0456\u0447\u0430\u0454\u0442\u044c\u0441\u044f \u0449\u043e\u0440\u0456\u0447\u043d\u043e 27 \u0432\u0435\u0440\u0435\u0441\u043d\u044f", "_v": 0}, {"date": "2021-06-08T20:15:27.396Z", "_id": "60bfd13eb1e28116a4edafb1", "title": "\u0414\u0435\u043d\u044c \u0412\u0438\u0448\u0438\u0432\u0430\u043d\u043a\u0438", "content": "\u0412\u0438\u0448\u0438\u0432\u0430\u043d\u043a\u0430 \u2013 \u043d\u043e\u0432\u0456\u0442\u043d\u044f \u043d\u0430\u0437\u0432\u0430 \u043d\u0430\u0440\u043e\u0434\u043d\u043e\u0456 \u0440\u0443\u0441\u044c\u043a\u043e\u0456 \u0441\u043e\u0440\u043e\u0447\u043a\u0438, \u043f\u0440\u0438\u043a\u0440\u0430\u0448\u0435\u043d\u043e\u0456 \u043e\u0440\u043d\u0430\u043c\u0435\u043d\u0442\u043e\u0432\u0430\u043d\u043e\u044e \u0432\u0438\u0448\u0438\u0432\u043a\u043e\u044e. \u0412\u0430\u0436\u043b\u0438\u0432\u0430 \u0441\u043a\u043b\u0430\u0434\u043e\u0432\u0430 \u0443\u043a\u0440\u0430\u0456\u043d\u0441\u044c\u043a\u043e\u0433\u043e \u043d\u0430\u0446\u0456\u043e\u043d\u0430\u043b\u044c\u043d\u043e\u0433\u043e \u0432\u0431\u0440\u0430\u043d\u043d\u044f. \u041f\u043e\u0434\u0456\u043b\u044f\u0442\u044c\u0441\u044f \u043d\u0430 \u0436\u0456\u043d\u043e\u0447\u0456 \u0456 \u0447\u043e\u043b\u043e\u0432\u0456\u0447\u0456 \u0441\u043e\u0440\u043e\u0447\u043a\u0438. \u041c\u0430\u0454 \u0441\u043f\u0456\u043b\u044c\u043d\u0435 \u043f\u043e\u0445\u043e\u0434\u0436\u0435\u043d\u043d\u044f \u0437 \u0432\u0438\u0448\u0438\u0442\u0438\u043c\u0438 \u0441\u043e\u0440\u043e\u0447\u043a\u0430\u043c\u0438 \u0456 \u0441\u043b\u043e\u0432'\u044f\u043d\u0441\u044c\u043a\u0438\u043c\u0438 \u043d\u0430\u0440\u043e\u0434\u0456\u0432 \u0421\u0445\u0456\u0434\u043d\u043e\u0456 \u0456 \u0426\u0435\u043d\u0442\u0440\u0430\u043b\u044c\u043d\u043e\u0456 \u0415\u0432\u0440\u043e\u043f\u0438. \u0412\u0456\u0434 \u0447\u0430\u0441\u0456\u0432 \u0441\u0435\u0440\u0435\u0434\u043d\u044c\u043e\u0432\u0456\u0447\u0447\u044f \u0442\u0440\u0430\u0434\u0438\u0446\u0456\u0439\u043d\u043e \u043d\u043e\u0441\u0438\u043b\u0430\u0441\u044f \u044f\u043a \u043f\u043e\u0432\u0441\u044f\u043a\u0434\u0435\u043d\u043d\u0456\u0439 \u0456 \u0441\u0432\u044f\u0442\u043a\u043e\u0432\u0438\u0439 \u043e\u0434\u044f\u0433. \u0417 XX \u0441\u0442\u043e\u043b\u0456\u0442\u0442\u044f \u043d\u043e\u0441\u0438\u0442\u044c\u044f \u043e\u043a\u0440\u0435\u043c\u043e, \u0440\u0430\u0437\u043e\u043c \u0437 \u043a\u043e\u0441\u0442\u044e\u043c\u043e\u043c, \u044f\u043a \u0441\u0432\u044f\u0442\u043a\u043e\u0432\u0438\u0439, \u043f\u0430\u0442\u0440\u0456\u043e\u0442\u0438\u0447\u043d\u0438\u0439 \u0456 \u043a\u0443\u043b\u044c\u0442\u043e\u0432\u0438\u0439 \u043e\u0434\u044f\u0433. \u0414\u0440\u0430\u0434\u0438\u0446\u0456\u0439\u043d\u0430 \u0432 \u0432\u0438\u0448\u0438\u0432\u043a\u0430 \u0441\u043e\u0440\u043e\u0447\u043e\u043a \u043c\u0430\u0454 \u043e\u0431\u0435\u0440\u0435\u0433\u043e\u0432\u0435 \u0437\u043d\u0430\u0447\u0435\u043d\u043d\u044f \u0456 \u0440\u0456\u0437\u043d\u0438\u0442\u044c\u044f\u0441\u044f \u0437\u0430\u043b\u0435\u0436\u043d\u043e \u0432\u0456\u0434 \u0440\u0435\u0433\u0456\u043e\u043d\u0443", "_v": 0}]\nGET /api 200 3180 - 813.481 ms
```

Рисунок 3.25 — Отримання даних з БД

В лістингу 3.2 наведено код підключення до mongoDB.

### Лістинг 3.2 – Підключення до mongoDB.

```
const app = express();
const PORT = process.env.PORT || 8080;
const MONGODB_URL = 'mongodb+srv://admin:admin@news.ykouq.mongodb.net/News?retryWrites=true&w=majority'
```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 41   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

## Продовження лістингу 3.2

```
mongoose.connect(MONGODB_URL, {
  useNewUrlParser: true,
  useUnifiedTopology: true
});

mongoose.connection.on('connected', () =>{
  console.log('Mongoose is conected!!');
});
```

Також нам потрібно і записувати дані в базу, код наведений в лістингу 3.3.

### Лістинг 3.3 – Запис в БД.

```
router.post('/save', (req, res) => {
  console.log("Body: ", req.body);
  const data = req.body;
  const newRec = new New(data);

  newRec.save((error) => {
    if (error) {
      res.status(500).json({ msg: "Sorry, server problems" });
      return;
    }
    return res.json({
      msg: "We received your data"
    });
  });
});
```

Для додавання медіафайлу в статтю переносимо його в папку сайту (рисунк 3.26).

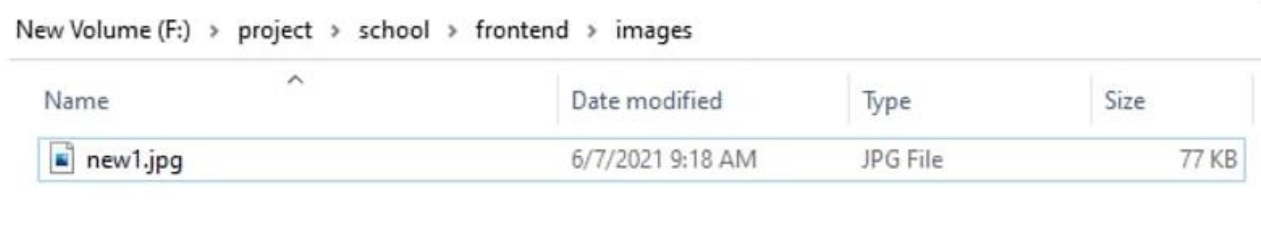


Рисунок 3.26 — Додавання фото в папку сайту

При створенні статті в адмін-панелі натискаємо кнопку вибрати зображення (рисунк 3.27).

← Повернутися назад

Виберіть зображення

Введіть заголовок статті

Введіть вміст статті

Зберегти

Рисунок 3.27 — Створення статті

Потім відкривається вікно із доступними медіафайлами(рисунок 3.28), вибираємо потрібний як зображено на рисунку 3.29.

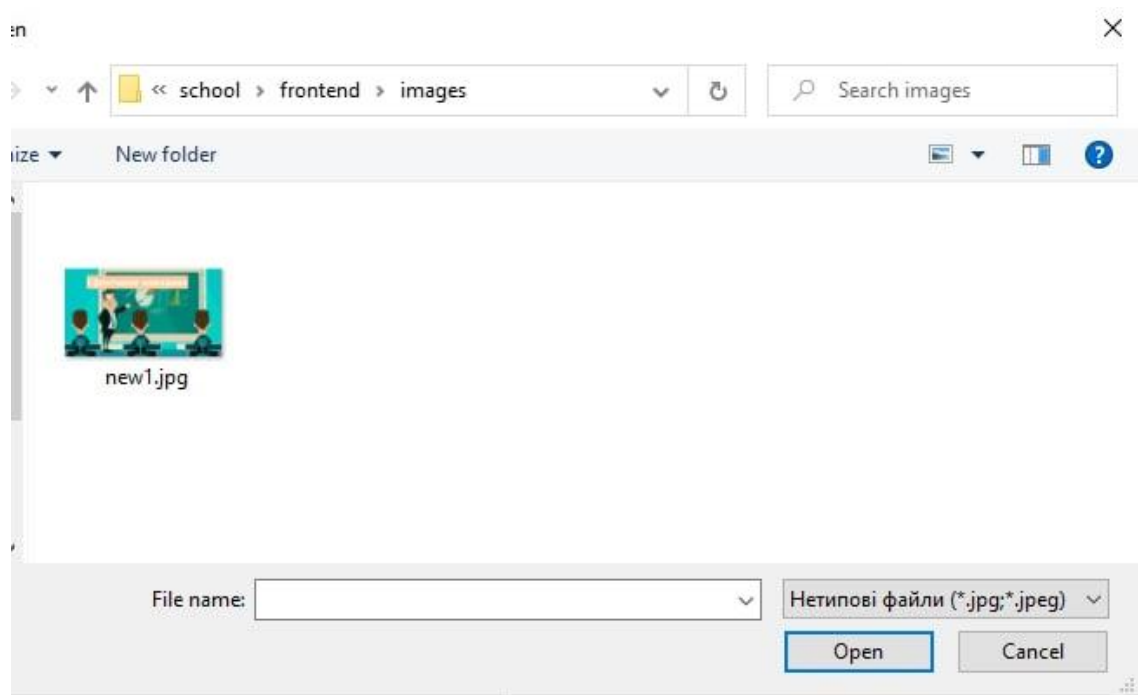


Рисунок 3.28 — Вікно з медіафайлами

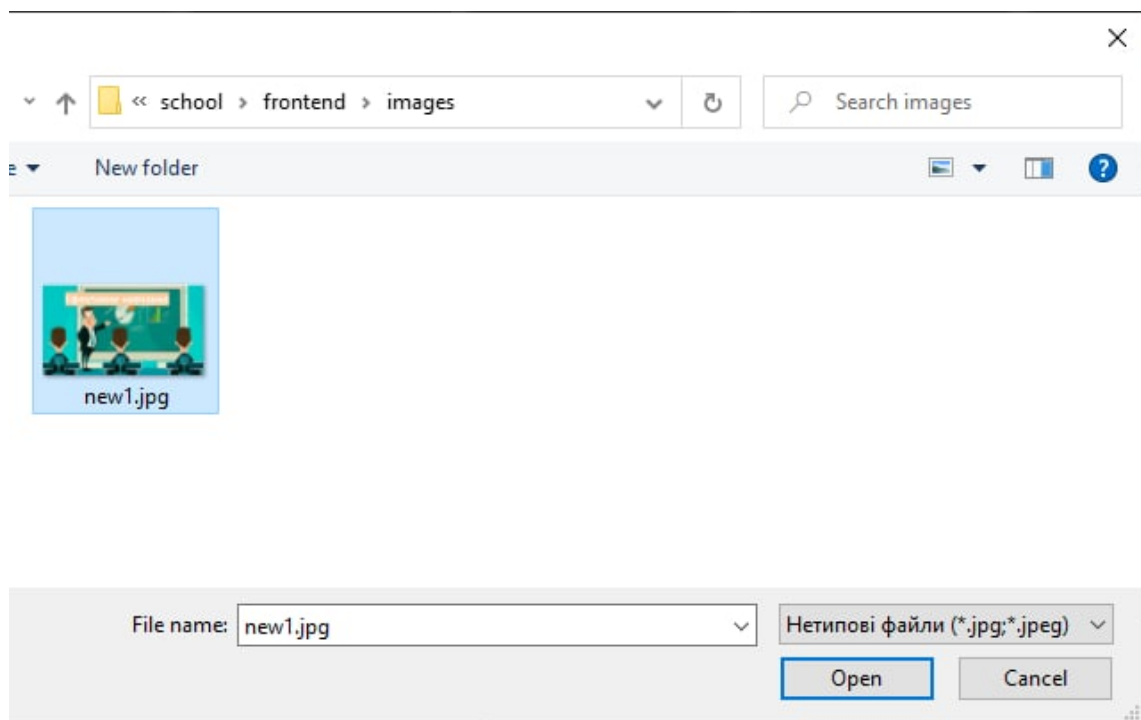


Рисунок 3.29 — Вибір фото

Після цих дії буде повідомлення про успішне додавання зображення  
 рисунок 3.30.

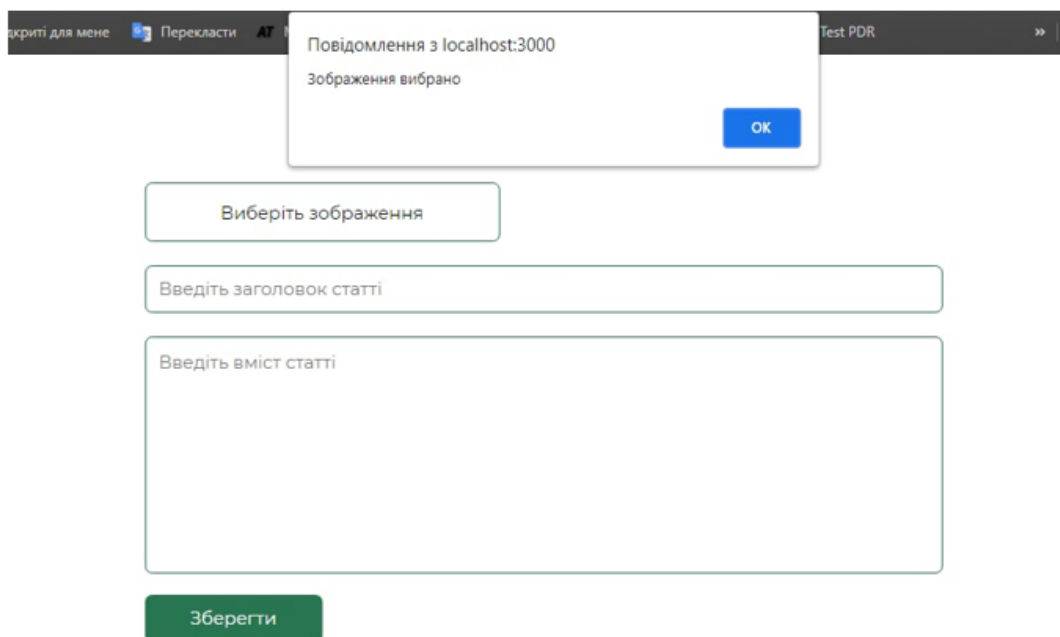
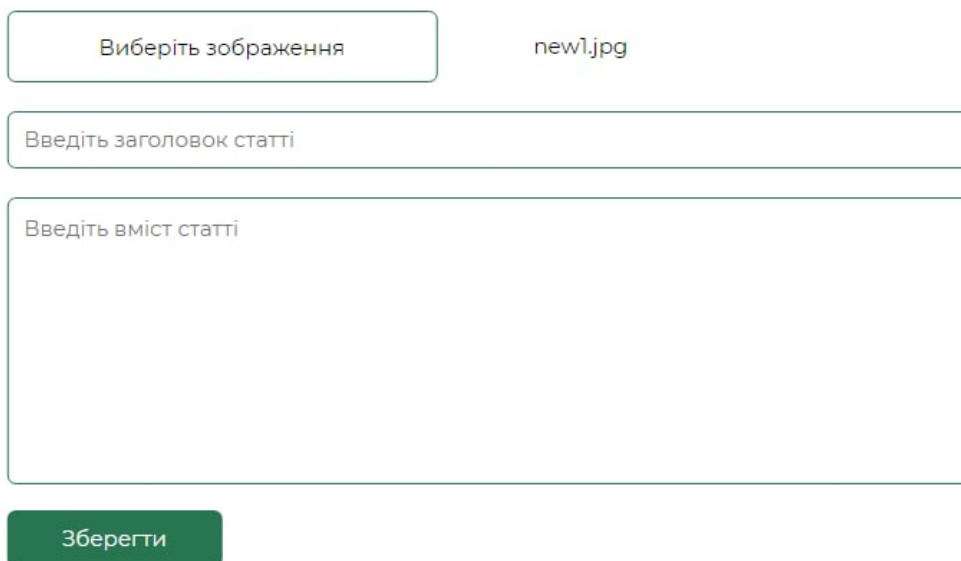


Рисунок 3.30 — Повідомлення про додане зображення

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 44   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

Після натискання на кнопку «ОК» з'явиться назва доданого зображення(рисунок3.31).



The screenshot shows a web interface for adding content. At the top, there is a button labeled "Виберіть зображення" (Select image) and the filename "new1.jpg" is displayed next to it. Below this is a text input field labeled "Введіть заголовок статті" (Enter article title). Underneath the title field is a larger text area labeled "Введіть вміст статті" (Enter article content). At the bottom of the form is a green button labeled "Зберегти" (Save).

Рисунок 3.31 — Успішно додане зображення

Прописуємо заголовок та саму статтю та натискаємо кнопку «Зберегти», після цього з'явиться повідомлення про те що стаття додана(рисунок 3.32).

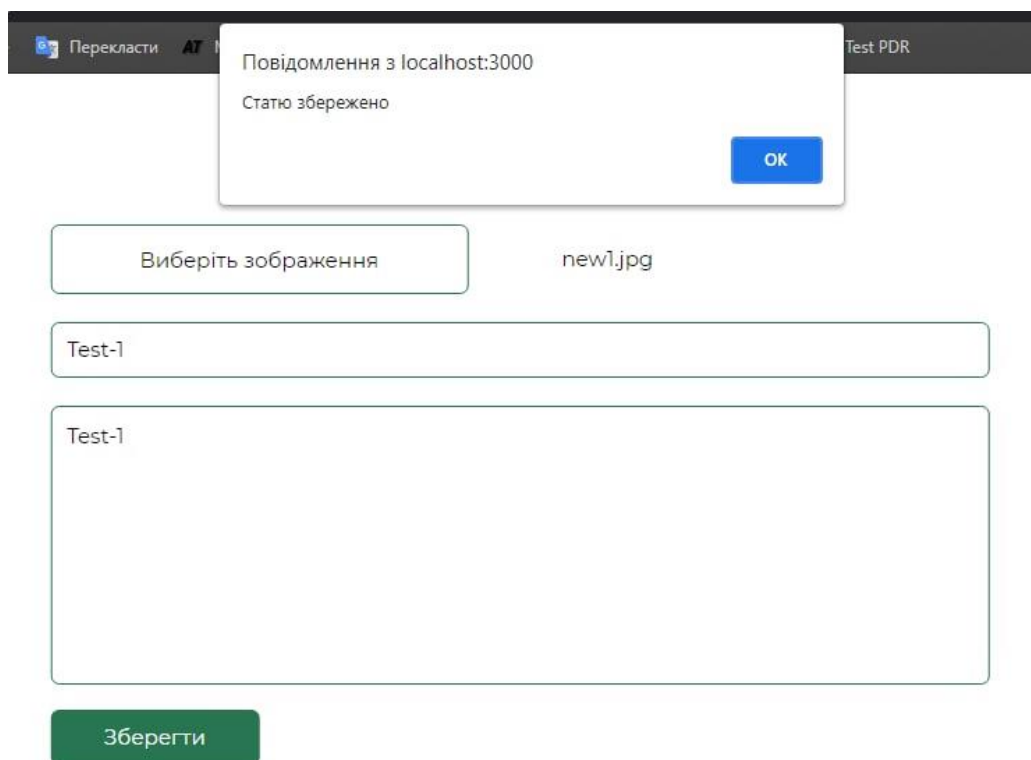


Рисунок 3.32 — Повідомлення про додану статтю

Вона успішно додалася на сайт, зображення на рисунку 3.33.

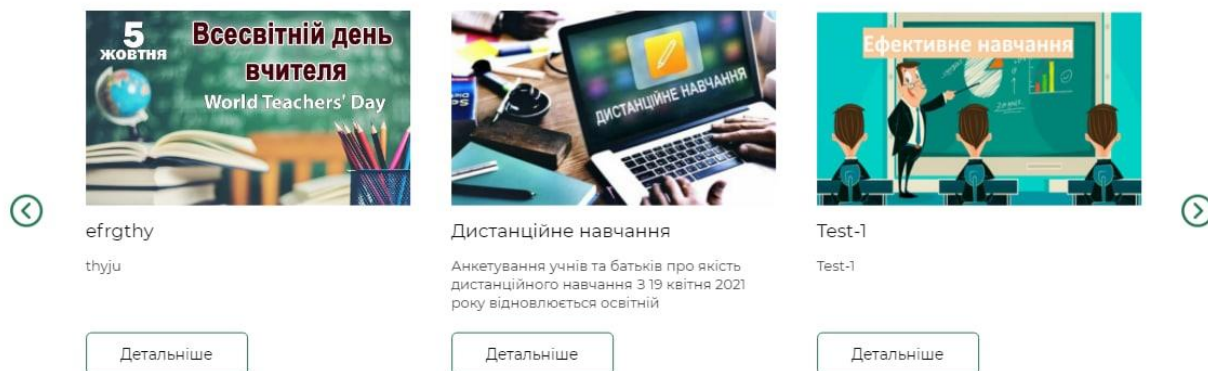


Рисунок 3.33 — Додана стаття на веб-сайті

Для того щоб видалити статтю потрібно так само перейти в адмін-панель як зображено на рисунку 3.34.

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 46   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

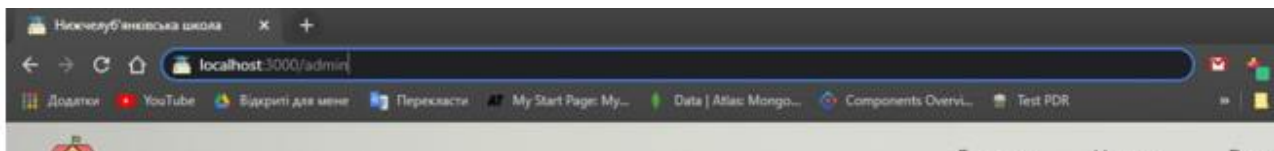


Рисунок 3.34 —Перехід в адмін-панель

Обираємо статтю яку хочемо видалити та нитискаємо біля неї на червоний хрестик(рисунок 3.35), після цього з'явиться відповідне повідомлення.

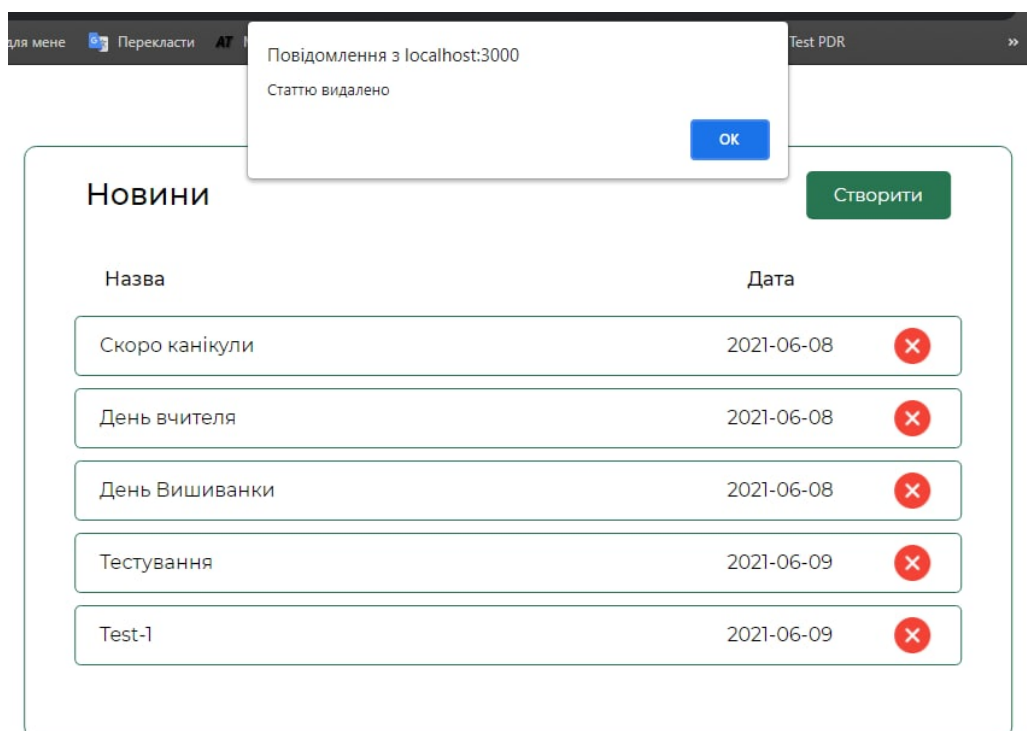


Рисунок 3.35 —Видалення статті

Тиснемо ОК і все, стаття видалена, результат на рисунку 3.36.



Новини

Створити

| Назва          | Дата       |   |
|----------------|------------|---|
| Скоро канікули | 2021-06-08 | ✕ |
| День вчителя   | 2021-06-08 | ✕ |
| День Вишиванки | 2021-06-08 | ✕ |
| Тестування     | 2021-06-09 | ✕ |

Рисунок 3.36 — Успішне видалення статті

Макет веб-сайту був спроектований в програмі Figma. Figma – це програма для розробки інтерфейсів, яка працює як в браузері, так і десктопній версії. Figma надає всі інструменти, необхідні для етапу проектування, включаючи векторні інструменти, які здатні до повноцінної ілюстрації, а також можливості складання прототипів та створення коду для передачі даних. Програму зображено на рисунку 3.37.

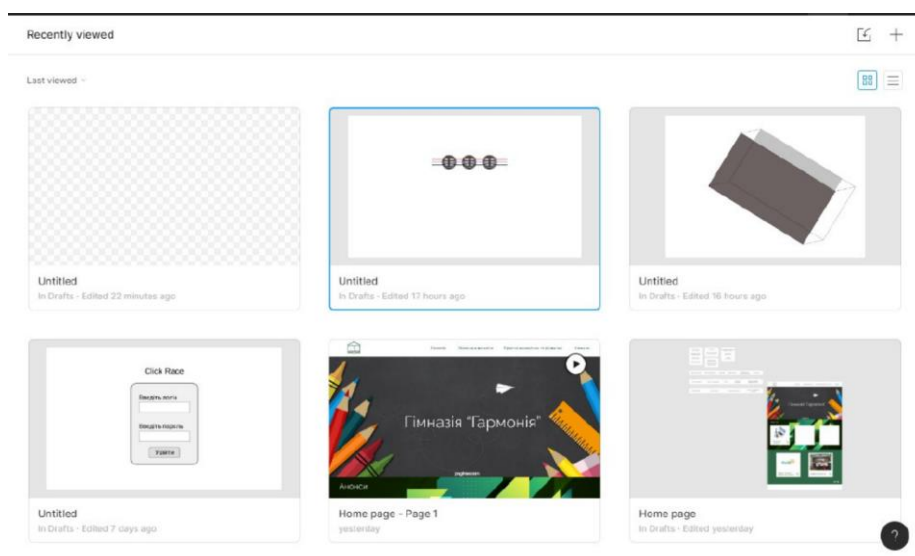


Рисунок 3.37 — Початкове вікно програми Figma

На рисунку 3.38 та 3.39 зображено процес формування макета.

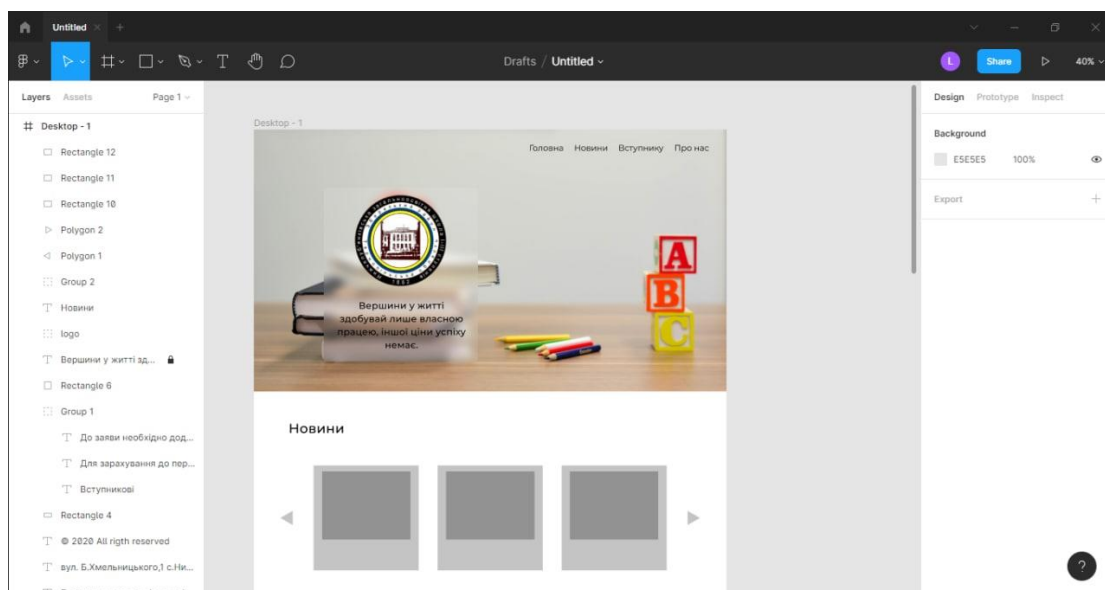


Рисунок 3.38 — Створення макету сайту

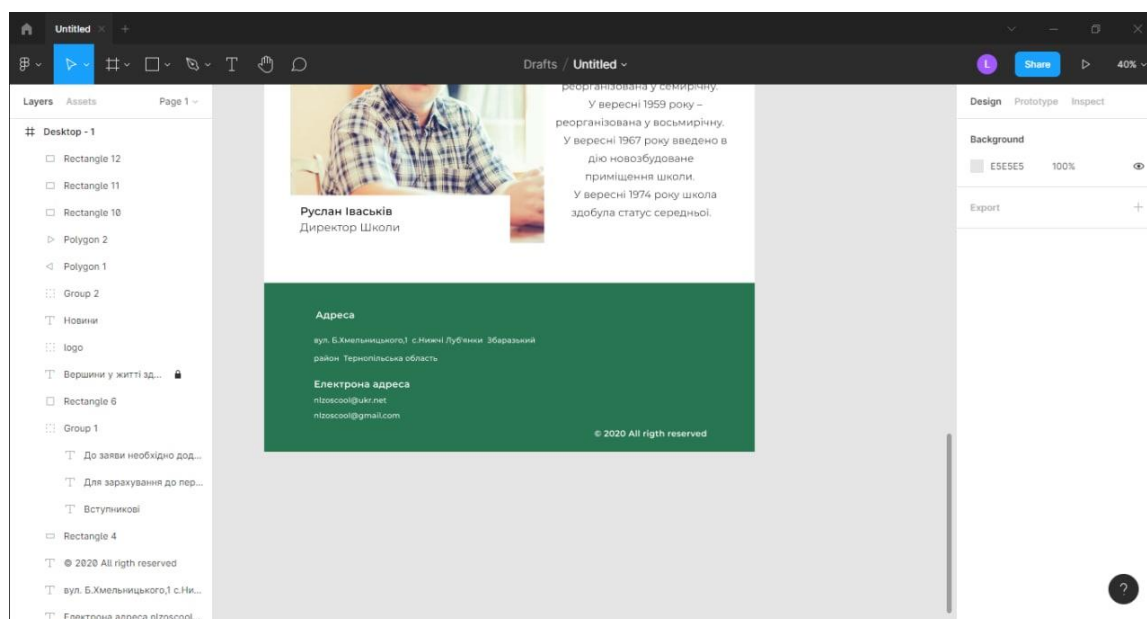


Рисунок 3.39 — Компоненти Figma

Протестуємо роботу сайту зі сторони користувача.

При переході на веб-сайт користувач потрапляє на головну сторінку сайту і може пересуватися по пунктах меню.

Переглядаючи новини користувач може користуватися стрілками слайдера (рисунок 3.40).

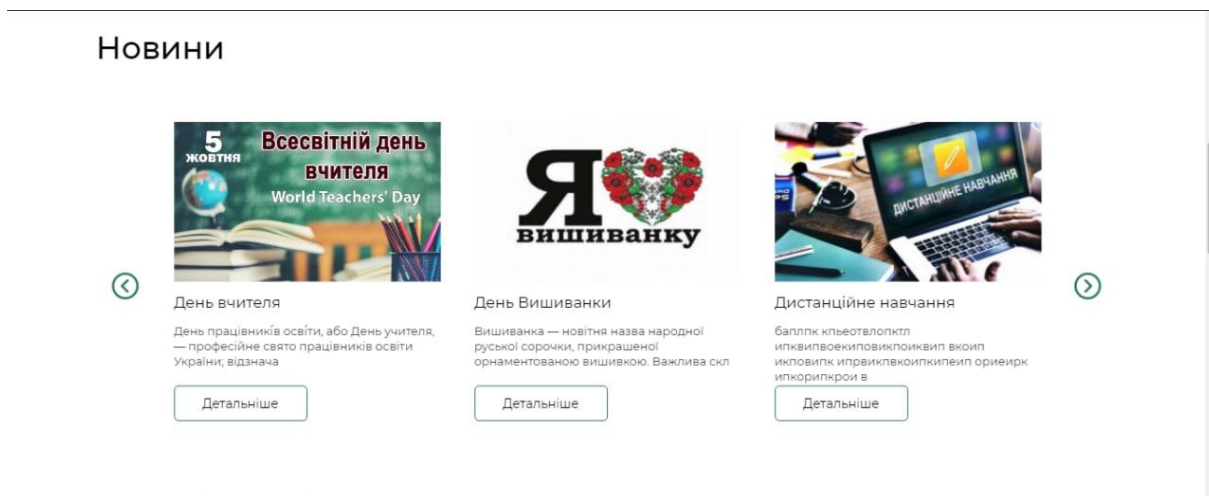


Рисунок 3.40 — Взаємодія із слайдером

Знайшовши цікаву статтю користувач може переглянути її детальніше натиснувши кнопку «Детальніше», як зображено на рисунку 3.41.

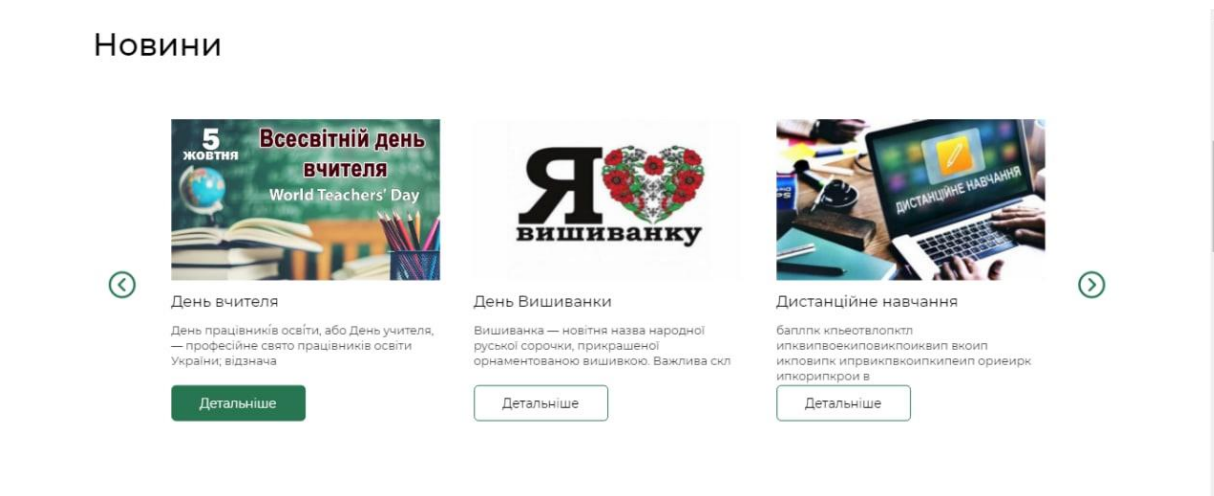


Рисунок 3.41 — Перехід до статті

Після переходу користувач потрапляє на сторінку статті та може ознайомлюватися з нею в повному розмірі (рисунки 3.42 та 3.43).



Рисунок 3.42 — Відкрита стаття

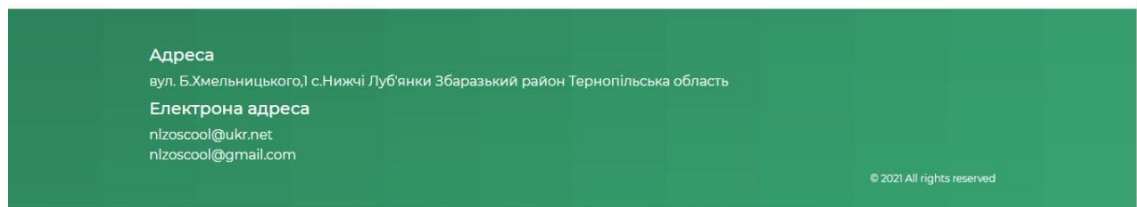


Рисунок 3.43 — Відкрита стаття

Всі дані успішно додаються як на сайт так і в базу даних та видалається, додаток працює без помилок чи інших «негараздів».

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 51   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

## 4 ТЕХНІКО-ЕКОНОМІЧНЕ ОБҐРУНТУВАННЯ

### 4.1 Аналіз ринку

Метою дипломного проєкту є – створення web-сайту школи, що розташований у с. Нижчелуб'янськ. Web-сайт буде розміщено у всесвітній мережі Інтернет, доступ до якого буде доступний з усіх платформ та усіх операційних систем за допомогою браузерів. Сайт розроблено на платформі React.js. Створення даного сайту дозволить розміщувати головні новини школи в реальному часі швидко і просто, а також ділитися з усіма бажаючими різними статтями та повсякденним життям школи.

Створений web-сайт повністю задовільняє потреби замовника, сайт легкий в редагуванні, а зручний та красивий інтерфейс робить його привабливим для відвідувача. Сайт буде розміщено на хостингу та буде придбане доменне ім'я за кошти селищної ради, подальша підтримка та оновлення сайту буде здійснюватися розробником, проте в майбутньому права адміністратора можуть бути передані іншій особі.

### 4.2 Розрахунок витрат на проєктування

#### Зарплата розробника

Статтею 6 Закону про оплату праці і ст. 96 КЗпП установлено, що розмір мінімального посадового окладу (тарифної ставки) працівника не має бути менший ПМПО на 1 січня календарного року. У 2021 році такий мінімум складає 6000 грн., а отже, і мінімальний розмір окладу на увесь 2021 рік має бути не нижчий 6000 грн.

Адміністратору сайту нараховані за повний відпрацьований місяць 7520 грн. Податкова соціальна пільга до такої заробітної плати не застосовується, оскільки вона більша за граничний розмір доходу, який дає право на податкову соціальну пільгу.

1) Рахуємо податок на доходи фізичних осіб:  $7520 \times 18\%$  (ставка податку на доходи фізичних осіб) = 1 353.6 грн.

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 52   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

2) Рахуємо військовий збір:  $7520 \times 1,5\%$  (ставка військового збору) = 112,8 грн.

3) Рахуємо єдиний внесок:  $7550 \times 22\%$  (ставка ЄСВ) = 1654,4 грн.

4) Утримання – 1 466,4 грн. (1 353,6 грн. + 112,8 грн.)

5) До виплати працівникові – 6 053,6 грн.

7520 грн. – 1 353,6 грн. – 112,8 грн.

Розробнику web-сайту нараховані за повний відпрацьований місяць 8210 грн. Податкова соціальна пільга до такої заробітної плати не застосовується, оскільки вона більша за граничний розмір доходу, який дає право на податкову соціальну пільгу.

1) Рахуємо податок на доходи фізичних осіб:  $8210 \times 18\%$  (ставка податку на доходи фізичних осіб) = 1 477,8 грн.

2) Рахуємо військовий збір:  $8210 \times 1,5\%$  (ставка військового збору) = 123,15 грн.

3) Рахуємо єдиний внесок:  $8210 \times 22\%$  (ставка ЄСВ) = 1806,2 грн.

4) Утримання – 1600,95 грн. (1 477,8 грн. + 123,15 грн.)

5) До виплати працівникові – 6 609,05 грн.

8210 грн. – 1477,8 грн. – 123,15 грн.

Розрахування заробітної плати подано у таблиці 4.1

Таблиця 4.1 – Розрахування заробітної плати

| Посада        | Оклад             | Відрахування    | Кількість |        | Сума            |
|---------------|-------------------|-----------------|-----------|--------|-----------------|
| Розробник ПЗ  | 8210<br>грн./міс. | 1600,95 грн/міс | 1 чол.    | 1 міс. | 6609,05грн.     |
| Адміністратор | 7520<br>грн./міс. | 1466,4 грн/міс  | 1 чол.    | 1 міс. | 6053,6 грн.     |
|               | Усього зарплати:  |                 |           |        | 12663,1<br>грн. |

Контрагентські роботи не проводилися та працівники не були у відрядженні, тому витрат на це немає.

Інші прямі витрати за місяць 350,9 грн (матеріали, обладнання). Усього прямих витрат 350,9 грн. – витрати на компоненти для проєкту:

- 200 грн. (хостинг);
- 50,9 грн. (домен);
- 100 грн. (супутні матеріали).

Накладні витрати за місяць становлять 200 грн.

Планові накопичення у відсотках (25-35) від суми прямих і накладних витрат.

Планові накопичення за місяць становлять 137,7 грн  $((200+350,9) \times 0,25)$ .

Усього кошторисна вартість проєкту 688,6 грн. (сума прямих і накладних витрат та планових накопичень).

#### 4.3 Обґрунтування необхідності розробки

Даний web-сайт був розроблений для магазину Нижчелуб'янківської школи, що розташована на території с. Нижчелуб'янківськ, Цей сайт призначений для публікування актуальних новин та події що відбуваються на території школи та за її участі.

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 54   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

## ВИСНОВКИ

На підставі проведеного аналізу предметної області було поставлено задачу та мету web-сайту школи, визначені та описані функції які в ньому мають функціонувати.

Було спроектовано з «нуля» унікальний шаблон веб-сайту, у якому передбачено в онлайн режимі додавати та видаляти новини та статті із медіафайлами.

Наступним завданням було визначення засобів для реалізації поставленого завдання. Тому було вирішено обрати технологію розробки на платформі Node.js та базу даних сайту MongoDB. Також визначено мову програмування JS бібліотеку React та середовище розробки Visual Studio Code.

В результаті було реалізовано та протестовано web-сайт, який описувався у розділах цього звіту.

В подальшому буде розширений функціонал сайту, задіявши нові технології розробки, а також планується реалізувати проведення електронного навчання під час карантину.

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 55   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |



## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Створюємо новий React-додаток. *React*: веб-сайт. URL: <https://uk.reactjs.org/docs/create-a-new-react-app.html> (дата звернення: 20.02.2021)
2. Установка и начало работы с MongoDB на Windows. *Metanit.com*: веб-сайт. URL: <https://metanit.com/nosql/mongodb/1.2.php> (дата звернення: 25.02.2021).
3. Create React App with MongoDB. *ObjectRocket*: веб-сайт. URL: <https://kb.objectrocket.com/mongo-db/create-react-app-with-mongodb-part-4-connecting-backend-with-the-frontend-902> (дата звернення: 10.03.2021).
4. Нижчелуб'янківська школа. *sites.google.com*: веб-сайт. URL: <https://sites.google.com/site/nizcelubankivskashkola/home> (дата звернення: 15.03.2021).
5. MongoDB Atlas. *mongoDB*: веб-сайт. URL: <https://www.mongodb.com/cloud/atlas> (дата звернення: 14.04.2021).
6. Сільський ревізор Нижчі Луб'янки. *agropolit.com*: веб-сайт. URL: <https://agropolit.com/selskij-revizor/161-silskiy-revizor-nijchi-lubyanki> (дата звернення: 26.04.2021).
7. NPM. Package.json. Установка модулей. Определение команд. *Metanit.com*: веб-сайт. URL: <https://metanit.com/web/nodejs/2.4.php> (дата звернення: 18.03.2021).
8. Free vector icons – SVG, PSD, PNG. *flaticon*: веб-сайт. URL: <https://www.flaticon.com/> (дата звернення: 14.04.2021).

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 56   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

## ДОДАТКИ

### Додаток А

#### Файл **About\_Us.js**

```
import React from 'react'
import styles from './About_Us.module.css';

const About_Us = () =>{
  return(
    <div className={styles.main} id="about_us">
      <p className={styles.main__title}>Про нас</p>
    </div>
  );
}
export default About_Us;
```

#### Файл **About\_Us.module.css**

```
.main{
  width: 100%;
  height: 13rem;
  background:linear-gradient(180deg, rgba(0, 0, 0, 0.34)
0%, rgba(0, 0, 0, 0.68) 100%), url("./back.jpg");
  background-position: center;
}
.main .main__title{
  color: #fff;
  line-height: 13rem;
  text-align: center;
  font-family: mon-medium;
  font-size: 65px;
  letter-spacing: 1.5px;
}
```

#### Файл **Details.js**

```
import React from 'react';
import styles from './Details.module.css';

const Details = () =>{
  return(
    <div className={styles.details}>
      <div className={styles.content}>
        <div className={styles.director}>
          <div className={styles.info}>
            <p
className={styles.info__name}>Руслан Іваськів</p>
            <p
className={styles.info__position}>Директор Школи</p>
          </div>
        </div>
      <div className={styles.history}>
```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 57   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

```

        <p className={styles.text}>
            У 1882 році у селі Нижчі Луб'янки
була створена початкова школа.<br/>
            У 1945 році школа була реорганізована
у семирічну.<br/>
            У вересні 1959 року – реорганізована
у восьмирічну.<br/>
            У вересні 1967 року введено в дію
новозбудоване приміщення школи.<br/>
            У вересні 1974 року школа здобула
статус середньої.

```

```

        </p>
    </div>
</div>
</div>
);
}

```

```
export default Details;
```

#### Файл **Details.module.css**

```

.details{
    width: 100%;
    height: 48rem;
    position: relative;
}
.details .content{
    width: 75%;
    height: 38rem;
    position: absolute;
    top: 50%;
    left: 50%;
    transform: translate(-50%,-50%);
    display: flex;
    justify-content: space-between;
}
.details .content .director{
    width: 500px;
    height: 100%;
    background: url('./director.jpg');
    background-position: 50%;
    background-repeat: no-repeat;
    background-size: 195% 100%;
    position: relative;
}
.content .director .info{
    width: 430px;
    height: 80px;
    position: absolute;
    background: #fff;
    bottom: 0;
    left: 0;
}

```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 58   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

```

.director .info .info__name{
  font-family: mon-medium;
  font-size: 24px;
  margin-left: 30px;
  margin-top: 15px;
  line-height: 30px;
}
.director .info .info__position{
  font-family: mon-reg;
  font-size: 18px;
  margin-left: 30px;
  line-height: 28px;
}
.details .content .history{
  height: 100%;
  width: 420px;
  display: block;
  margin-right: 50px;
}
.content .history .text{
  font-family: mon-reg;
  font-size: 25px;
  text-align: center;
  line-height: 45px;
}

```

### Файл Entrant.js

```

import React from 'react';
import styles from './Entrant.module.css';
import books from './books.png';

const Entrant = () =>{
  return(
    <div className={styles.main} id="entrant">
      <p className={styles.main__title}>Вступнику</p>
      <div className={styles.mainContent}>
        <p className={styles.mainContent__text}>
          Для зарахування до першого класу
необхідно надати заяву про
          зарахування за зразком, встановленим
додатком 1 до Порядку
          зарахування учнів, визначеного
Міністерством освіти.
          <br/>
          <br/>
          До заяви необхідно додати обов'язкові
документи:
          </p>
          <ul className={styles.mainContent__doc}>
            <li>Копія свідоцтва про народження</li>
            <li>Оригінал медичної довідки</li>
            <li>Документ, що підтверджує місце
проживання</li>

```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 59   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

```

        </ul>
    </div>
    <img src={books} alt="Документи"
className={styles.main__books} />
</div>

);
}

```

export default Entrant;

### Файл **Entrant.module.css**

```

.main{
    width: 100%;
    height: 40rem;
    background-color: #287551;
    background-image: url("data:image/svg+xml,%3Csvg
xmlns='http://www.w3.org/2000/svg' width='100%25' height='100%25'
viewBox='0 0 1200 800'%3E%3Cdefs%3E%3ClinearGradient id='a'
gradientUnits='userSpaceOnUse' x1='600' y1='25' x2='600'
y2='777'%3E%3Cstop offset='0' stop-color='%23287551'/%3E%3Cstop
offset='1' stop-
color='%23349668'/%3E%3C/linearGradient%3E%3ClinearGradient id='b'
gradientUnits='userSpaceOnUse' x1='650' y1='25' x2='650'
y2='777'%3E%3Cstop offset='0' stop-color='%23297853'/%3E%3Cstop
offset='1' stop-
color='%23359869'/%3E%3C/linearGradient%3E%3ClinearGradient id='c'
gradientUnits='userSpaceOnUse' x1='700' y1='25' x2='700'
y2='777'%3E%3Cstop offset='0' stop-color='%232a7b55'/%3E%3Cstop
offset='1' stop-
color='%2335996a'/%3E%3C/linearGradient%3E%3ClinearGradient id='d'
gradientUnits='userSpaceOnUse' x1='750' y1='25' x2='750'
y2='777'%3E%3Cstop offset='0' stop-color='%232b7e57'/%3E%3Cstop
offset='1' stop-
color='%23369b6b'/%3E%3C/linearGradient%3E%3ClinearGradient id='e'
gradientUnits='userSpaceOnUse' x1='800' y1='25' x2='800'
y2='777'%3E%3Cstop offset='0' stop-color='%232c8159'/%3E%3Cstop
offset='1' stop-
color='%23369d6c'/%3E%3C/linearGradient%3E%3ClinearGradient id='f'
gradientUnits='userSpaceOnUse' x1='850' y1='25' x2='850'
y2='777'%3E%3Cstop offset='0' stop-color='%232d845b'/%3E%3Cstop
offset='1' stop-
color='%23379e6d'/%3E%3C/linearGradient%3E%3ClinearGradient id='g'
gradientUnits='userSpaceOnUse' x1='900' y1='25' x2='900'
y2='777'%3E%3Cstop offset='0' stop-color='%232f875e'/%3E%3Cstop
offset='1' stop-
color='%2337a06f'/%3E%3C/linearGradient%3E%3ClinearGradient id='h'
gradientUnits='userSpaceOnUse' x1='950' y1='25' x2='950'
y2='777'%3E%3Cstop offset='0' stop-color='%23308a60'/%3E%3Cstop
offset='1' stop-
color='%2338a170'/%3E%3C/linearGradient%3E%3ClinearGradient id='i'
gradientUnits='userSpaceOnUse' x1='1000' y1='25' x2='1000'
y2='777'%3E%3Cstop offset='0' stop-color='%23318d62'/%3E%3Cstop
offset='1' stop-

```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 60   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

```

color='%2338a371'/%3E%3C/linearGradient%3E%3ClinearGradient id='j'
gradientUnits='userSpaceOnUse' x1='1050' y1='25' x2='1050'
y2='777'%3E%3Cstop offset='0' stop-color='%23329064'/%3E%3Cstop
offset='1' stop-
color='%2339a572'/%3E%3C/linearGradient%3E%3ClinearGradient id='k'
gradientUnits='userSpaceOnUse' x1='1100' y1='25' x2='1100'
y2='777'%3E%3Cstop offset='0' stop-color='%23339366'/%3E%3Cstop
offset='1' stop-
color='%2339a673'/%3E%3C/linearGradient%3E%3ClinearGradient id='l'
gradientUnits='userSpaceOnUse' x1='1150' y1='25' x2='1150'
y2='777'%3E%3Cstop offset='0' stop-color='%23349668'/%3E%3Cstop
offset='1' stop-
color='%233aa874'/%3E%3C/linearGradient%3E%3C/defs%3E%3Cg
%3E%3Crect fill='url(%23a)' width='1200' height='800'/%3E%3Crect
fill='url(%23b)' x='100' width='1100' height='800'/%3E%3Crect
fill='url(%23c)' x='200' width='1000' height='800'/%3E%3Crect
fill='url(%23d)' x='300' width='900' height='800'/%3E%3Crect
fill='url(%23e)' x='400' width='800' height='800'/%3E%3Crect
fill='url(%23f)' x='500' width='700' height='800'/%3E%3Crect
fill='url(%23g)' x='600' width='600' height='800'/%3E%3Crect
fill='url(%23h)' x='700' width='500' height='800'/%3E%3Crect
fill='url(%23i)' x='800' width='400' height='800'/%3E%3Crect
fill='url(%23j)' x='900' width='300' height='800'/%3E%3Crect
fill='url(%23k)' x='1000' width='200' height='800'/%3E%3Crect
fill='url(%23l)' x='1100' width='100'
height='800'/%3E%3C/g%3E%3C/svg%3E");
    background-attachment: fixed;
    background-size: cover;
    color: #fff;
    position: relative;
}

.main__title{
    font-size: 35px;
    font-family: mon-medium;
    padding: 70px 0 0 100px;
}

.main .main__books{
    height: 200px;
    position: absolute;
    bottom: 15%;
    right: 15%;
}

.main .mainContent{
    width: 75%;
    font-family: mon-reg;
    position: absolute;
    top: 58%;
    left: 50%;
    transform: translate(-50%,-55%);
}

.mainContent .mainContent__text{
    line-height: 38px;

```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 61   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

```

        margin-bottom: 10px;
        font-size: 25px;
    }
    .mainContent .mainContent__doc{
        line-height: 35px;
        font-size: 21px;
    }
    .mainContent__doc li::before{
        content: "✓";
        margin-right: 10px;
        margin-left: 30px;
    }

```

### Файл Footer.js

```

import React from 'react';
import styles from './Footer.module.css';

const Footer = () =>{
    return(
        <div className={styles.footer}>
            <div className={styles.content}>
                <p
className={styles.content__title}>Адреса</p>
                <p className={styles.content__text}>вул.
Б.Хмельницького,1 с.Нижчі Луб'янки Збаразький район Тернопільська
область</p>
                <p
className={styles.content__title}>Електрона адреса</p>
                <p
className={styles.content__text}>nlzoscool@ukr.net<br/>nlzoscool@g
mail.com</p>
                <p className={styles.footer__righths}>© 2021
All rights reserved</p>
            </div>
        </div>
    );
}

export default Footer;

```

### Файл Footer.module.css

```

.footer{
    width: 100%;
    height: 16rem;
    position: relative;
    background-color: #287551;
    background-image: url("data:image/svg+xml,%3Csvg
xmlns='http://www.w3.org/2000/svg' width='100%25' height='100%25'
viewBox='0 0 1200 800'%3E%3Cdefs%3E%3ClinearGradient id='a'
gradientUnits='userSpaceOnUse' x1='600' y1='25' x2='600'
y2='777'%3E%3Cstop offset='0' stop-color='%23287551'/%3E%3Cstop
offset='1' stop-

```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 62   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

```

color='%23349668'/%3E%3C/linearGradient%3E%3ClinearGradient id='b'
gradientUnits='userSpaceOnUse' x1='650' y1='25' x2='650'
y2='777'%3E%3Cstop offset='0' stop-color='%23297853'/%3E%3Cstop
offset='1' stop-
color='%23359869'/%3E%3C/linearGradient%3E%3ClinearGradient id='c'
gradientUnits='userSpaceOnUse' x1='700' y1='25' x2='700'
y2='777'%3E%3Cstop offset='0' stop-color='%232a7b55'/%3E%3Cstop
offset='1' stop-
color='%2335996a'/%3E%3C/linearGradient%3E%3ClinearGradient id='d'
gradientUnits='userSpaceOnUse' x1='750' y1='25' x2='750'
y2='777'%3E%3Cstop offset='0' stop-color='%232b7e57'/%3E%3Cstop
offset='1' stop-
color='%23369b6b'/%3E%3C/linearGradient%3E%3ClinearGradient id='e'
gradientUnits='userSpaceOnUse' x1='800' y1='25' x2='800'
y2='777'%3E%3Cstop offset='0' stop-color='%232c8159'/%3E%3Cstop
offset='1' stop-
color='%23369d6c'/%3E%3C/linearGradient%3E%3ClinearGradient id='f'
gradientUnits='userSpaceOnUse' x1='850' y1='25' x2='850'
y2='777'%3E%3Cstop offset='0' stop-color='%232d845b'/%3E%3Cstop
offset='1' stop-
color='%23379e6d'/%3E%3C/linearGradient%3E%3ClinearGradient id='g'
gradientUnits='userSpaceOnUse' x1='900' y1='25' x2='900'
y2='777'%3E%3Cstop offset='0' stop-color='%232f875e'/%3E%3Cstop
offset='1' stop-
color='%2337a06f'/%3E%3C/linearGradient%3E%3ClinearGradient id='h'
gradientUnits='userSpaceOnUse' x1='950' y1='25' x2='950'
y2='777'%3E%3Cstop offset='0' stop-color='%23308a60'/%3E%3Cstop
offset='1' stop-
color='%2338a170'/%3E%3C/linearGradient%3E%3ClinearGradient id='i'
gradientUnits='userSpaceOnUse' x1='1000' y1='25' x2='1000'
y2='777'%3E%3Cstop offset='0' stop-color='%23318d62'/%3E%3Cstop
offset='1' stop-
color='%2338a371'/%3E%3C/linearGradient%3E%3ClinearGradient id='j'
gradientUnits='userSpaceOnUse' x1='1050' y1='25' x2='1050'
y2='777'%3E%3Cstop offset='0' stop-color='%23329064'/%3E%3Cstop
offset='1' stop-
color='%2339a572'/%3E%3C/linearGradient%3E%3ClinearGradient id='k'
gradientUnits='userSpaceOnUse' x1='1100' y1='25' x2='1100'
y2='777'%3E%3Cstop offset='0' stop-color='%23339366'/%3E%3Cstop
offset='1' stop-
color='%2339a673'/%3E%3C/linearGradient%3E%3ClinearGradient id='l'
gradientUnits='userSpaceOnUse' x1='1150' y1='25' x2='1150'
y2='777'%3E%3Cstop offset='0' stop-color='%23349668'/%3E%3Cstop
offset='1' stop-
color='%233aa874'/%3E%3C/linearGradient%3E%3C/defs%3E%3Cg
%3E%3Crect fill='url(%23a)' width='1200' height='800'/%3E%3Crect
fill='url(%23b)' x='100' width='1100' height='800'/%3E%3Crect
fill='url(%23c)' x='200' width='1000' height='800'/%3E%3Crect
fill='url(%23d)' x='300' width='900' height='800'/%3E%3Crect
fill='url(%23e)' x='400' width='800' height='800'/%3E%3Crect
fill='url(%23f)' x='500' width='700' height='800'/%3E%3Crect
fill='url(%23g)' x='600' width='600' height='800'/%3E%3Crect
fill='url(%23h)' x='700' width='500' height='800'/%3E%3Crect
fill='url(%23i)' x='800' width='400' height='800'/%3E%3Crect

```

|                        |      |          |         |      |      |
|------------------------|------|----------|---------|------|------|
| ДП.КН 21.466.11.000 ПЗ |      |          |         |      | Арк. |
|                        |      |          |         |      | 63   |
| Зм.                    | Арк. | № докум. | Підпис. | Дата |      |



```

fill='url(%23j)' x='900' width='300' height='800'/%3E%3Crect
fill='url(%23k)' x='1000' width='200' height='800'/%3E%3Crect
fill='url(%23l)' x='1100' width='100'
height='800'/%3E%3C/g%3E%3C/svg%3E");
    background-attachment: fixed;
    background-size: cover;
}
.footer .content{
    width: 75%;
    height: 10rem;
    position: absolute;
    top: 50%;
    left: 50%;
    transform: translate(-50%,-50%);
    color: #fff;
}

.content .content__title{
    font-family: mon-medium;
    font-size: 20px;
    margin-bottom: 10px;
}
.content .content__text{
    font-family: mon-reg;
    font-size: 16px;
    line-height: 24px;
    margin-bottom: 10px;
}
.content .footer__rights{
    font-family: mon-reg;
    font-size: 12px;
    text-align: right;
}

```

### Файл Header.js

```

import React from 'react';
import styles from './Header.module.css';
import logo from './school.png'
import { Link } from "react-scroll";

const Header = () =>{
    var prevScrollpos = window.pageYOffset;

    window.onscroll = function() {
        var custom_head =
document.getElementById("custom_header");
        var currentScrollPos = window.pageYOffset;
        if (prevScrollpos > currentScrollPos) {
            custom_head.style.top = "0";
        } else {
            custom_head.style.top = "-75px";
        }
    }
}

```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 64   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

```

        prevScrollpos = currentScrollPos;
        if(currentScrollPos > 300){
            custom_head.style.background = "#fff";
        }else{
            custom_head.style.background = "none";
        }
    }
    return(
        <div className={styles.main} id="custom_header">
            <img src={logo} alt="Icon"
className={styles.main__icon} />
            <ul className={styles.main__menu}>
                <li className={styles.main__item}><Link
                    activeClass="active"
                    to="home"
                    spy={true}
                    smooth={true}
                    duration={70}>Головна</Link></li>
                <li className={styles.main__item}><Link
                    activeClass="active"
                    to="news"
                    spy={true}
                    smooth={true}
                    duration={70}>Новини</Link></li>
                <li className={styles.main__item}><Link
                    activeClass="active"
                    to="entrant"
                    spy={true}
                    smooth={true}
                    duration={70}>Вступнику</Link></li>
                <li className={styles.main__item}><Link
                    activeClass="active"
                    to="about_us"
                    spy={true}
                    smooth={true}
                    duration={70}>Про нас</Link></li>
            </ul>
        </div>
    )
}

```

export default Header;

### Файл **Header.module.css**

```

.main{
    width: 100%;
    height: 65px;
    display: grid;
    grid-template-columns: 25% 75%;
    position: fixed;
    z-index: 1;
    transition: background-color .4s, opacity .5s, visibility
.5s, top 0.5s ease-out;
}

```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 65   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

```

}

.main__icon{
  width: 40px;
  height: 40px;
  display: block;
  margin: 12px 40px;
}

.main .main__menu{
  line-height: 64px;
  display: block;
  width: 470px;
  margin: 0 25px 0 auto;
}

.main__menu .main__item{
  display: inline;
  font-size: 16px;
  font-family: mon-reg;
  margin: 0 55px 0 0;
  position: relative;
  cursor: pointer;
}

.main__item:last-child{
  margin: 0px;
}

```

### Файл Home.js

```

import React from 'react';
import styles from './Home.module.css';
import schoolLogo from './school-logo.png'

const Home = () =>{
  return(
    <div className={styles.main} id="home">
      <aside className={styles.main__aside}>
        <img src={schoolLogo} alt="Шкільний логотип"
className={styles.main__logo} />
        <p className={styles.main__losung}>Вершини у
житті здобувай лише власною працею, іншої ціни успіху немає.</p>
      </aside>
    </div>
  )
}

export default Home;

```

### Файл Home.module.css

```

.main{
  width: 100%;

```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 66   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

```

        height: 620px;
        position: relative;
        background-image: url('./backg-image.jpg');
        background-repeat: no-repeat;
        background-position: 100% 80%;
        background-color: #318f63b9;
        background-size: 100% 140%;
    }

    .main .main__aside{
        position: absolute;
        width: 300px;
        height: 450px;
        top: 17%;
        left: 17%;
        border-radius: 8px;
        box-shadow: 0 25px 50px rgba(0,0,0,0.2);
        background: transparent;
        backdrop-filter: blur(10px);
        transition: 0.5s;
    }

    .main__aside .main__logo{
        width: 220px;
        height: 220px;
        display: block;
        margin: 35px auto;
    }

    .main__aside .main__losung{
        display: block;
        margin: 45px auto;
        width: 260px;
        font-size: 18px;
        line-height: 30px;
        font-family: mon-medium;
        text-align: center;
    }

```

### Файл News.js

```

import React from 'react';
import styles from './News.module.css';
import axios from 'axios';
import {Link} from 'react-router-dom';
import "slick-carousel/slick/slick.css";
import "slick-carousel/slick/slick-theme.css";
import Slider from "react-slick";

function NextArrow(props) {
    const { onClick } = props;
    return <div className={styles.nextArrow}
onClick={onClick}/>

```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 67   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

```

    }

    function PrevArrow(props) {
        const { onClick } = props;
        return <div
className={styles.prevArrow}onClick={onClick}/>
    }

    class News extends React.Component{

        state ={
            title:'',
            content:'',
            source:'',
            news: []
        };

        componentDidMount =()=>{
            this.getNew();
        };

        getNew =()=>{
            axios.get('http://localhost:8080/api')
                .then((responce) =>{
                    const data = responce.data;
                    this.setState({news: data});
                    console.log('Data received!');
                })
                .catch(()=>{
                    alert("Error data!");
                });
        };

        displayNew =(news)=>{
            if(!news.length) return null;
            return news.map((news, index)=>{
                <div key={index} className={styles.item}>
                    <div className={styles.item__content}>
                        <img className={styles.content_img}
src={"/images/"+ news.source} alt={news.title} />
                        <p
className={styles.content_title}>{news.title}</p>
                        <p
className={styles.content_text}>{news.content.substr(0,100)}</p>
                        <Link className={styles.content_btn}
to={"/new/"+ (news._id)}>Детальніше</Link>
                    </div>
                </div>
            ));
        };

        render(){
            const settings = {
                infinite: true,
                speed: 500,
                arrows:true,

```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 68   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

```

        slidesToShow: 3,
        slidesToScroll: 1,
        nextArrow: <NextArrow />,
        prevArrow: <PrevArrow />
      };
    return (
      <div className={styles.main} id="news">
        <p className={styles.main__title}>Новини</p>
        <div className={styles.mainCarousel}>
          <Slider {...settings}>
            {this.displayNew(this.state.news)}
          </Slider>
        </div>
      </div>
    );
  };
}

export default News;

```

### Файл **News.module.css**

```

.main{
  width: 100%;
  height: 40rem;
  position: relative;
}
.main .main__title{
  font-size: 35px;
  font-family: mon-medium;
  padding: 70px 0 0 100px
}
.main .mainCarousel{
  width: 75%;
  height: 22.5rem;
  position: relative;
  top: 25%;
  left: 50%;
  transform: translate(-50%,-25%);
  /* border: 1px solid slateblue; */
}

.main .mainCarousel .item{
  height: 22.5rem;
  /* border: 1px solid red; */
  position: relative;
}

.mainCarousel .item .item__content{
  width: 300px;
  height: 100%;
  position: absolute;
  left: 50%;
  transform: translate(-50%);
}

```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 69   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

```

        /* border: 1px solid blueviolet; */
    }
    .item .item__content .content_img{
        width: 100%;
        height: 180px;
    }
    .item__content .content_title{
        font-size: 17px;
        font-family: mon-reg;
        margin: 15px 0;
        width: 100%;
    }
    .item__content .content_text{
        font-size: 13px;
        font-family: mon-reg;
        color: #424242;
        line-height: 17px;
        height: 70px;
        width: 100%;
    }
    .item__content .content_btn{
        width: 150px;
        height: 40px;
        border: 1px solid #287551;
        border-radius: 5px;
        background-color: #fff;
        font-family: mon-reg;
        font-size: 14px;
        display: flex;
        justify-content: center;
        align-items: center;
        color: #000;
    }
    .item__content .content_btn:hover{
        background-color: #287551;
        color: #fff;
        transition: all 0.4s ease-out;
    }
    .nextArrow, .prevArrow{
        width: 30px;
        height: 30px;
        display: block;
        position: absolute;
        top: 50%;
        background: url('./next.png');
        background-repeat: no-repeat;
        background-position: 100%;
        background-size: cover;
        cursor: pointer;
    }
    .nextArrow{
        right: -5%;
    }
}

```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 70   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

```
.prevArrow{
  transform: rotate(180deg);
  left: -5%;
}
```

### Файл **index.html**

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <link rel="icon" href="%PUBLIC_URL%/favicon.png" />
    <meta name="viewport" content="width=device-width,
initial-scale=1" />
    <link rel="manifest" href="%PUBLIC_URL%/manifest.json" />
    <title>Нижчелюб'янківська школа</title>
  </head>
  <body>
    <div id="root"></div>
  </body>
</html>
```

### Файл **AddNew.js**

```
import React from 'react';
import styles from './AddNew.module.css';
import {Link} from 'react-router-dom';
import axios from 'axios';
import arrow from '../icons/arrow.png';
class AddNew extends React.Component{
  state ={
    title:'',
    content:'',
    source:'',
    news: []
  };
  componentDidMount =()=>{
    this.getNew();
  }
  getNew =()=>{
    axios.get('http://localhost:8080/api')
      .then((responce) =>{
        const data = responce.data;
        this.setState({news: data});
        console.log('Data received!');
      })
      .catch(()=>{
        alert("Error data!");
      });
  };
  handleChandge =(event)=>{
    const target = event.target;
    const name = target.name;
    const val = target.value;
```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 71   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |



```

        this.setState({
            [name]:val
        });
    };
    onSub =(event)=>{
        event.preventDefault();
        const payload = {
            title: this.state.title,
            content: this.state.content,
            source: this.state.source
        }
        axios({
            url:'http://localhost:8080/api/save',
            method:'POST',
            data:payload
        })
        .then(()=>{
            console.log("Data sent to server");
            this.resetInputs();
            this.getNew();
            alert("Статю збережено");
        })
        .catch(()=>{
            console.log("Serer error");
        });
    };
    resetInputs =()=>{
        this.setState({
            title:'',
            content:'',
            source:''
        });
    };
    fileSelect = event =>{
        console.log(event.target.files[0].name);
        this.setState({
            source:event.target.files[0].name
        });
        alert("Зображення вибрано")
    }

    render() {
        console.log('State:', this.state);
        return (
            <div className={styles.addNew}>
                <Link to="/admin" className={styles.back}>
                    <img className={styles.arrow} src={arrow}
alt="Назад" />
                    Повернутися назад</Link>
                <div className={styles.input_form}>
                    <div className={styles.image}>
                        <label required id="f1"
className={styles.selectLabel}>Виберіть зображення

```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 72   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

```

        <input required id="file"
className={styles.selectImg} type="file"
onChange={this.fileSelect} accept=".jpg, .jpeg" />
        </label>
        <p
className={styles.fileName}>{this.state.source}</p>
        </div>
        <form autoComplete="off"
onSubmit={this.onSubmit}>
            <input onChange={this.handleChange}
value={this.state.title}
type="text"
name="title"
id="title"
className={styles.title}
placeholder="Введіть заголовок статті"
required />
            <textarea onChange={this.handleChange}
value={this.state.content}
name="content"
id="content"
className={styles.content}
placeholder="Введіть вміст статті"
required
/></textarea><br/>
            <input type="submit" value="Зберегти"
className={styles.btn} />
        </form>
    </div>
</div>
    );
};
}

export default AddNew;

```

### Файл **AddNew.module.css**

```

.addNew{
    width: 100%;
    height: 100vh;
    position: relative;
}

.input_form{
    width: 50%;
    height: 30rem;
    position: absolute;
    top: 50%;
    left: 50%;
    transform: translate(-50%,-50%);
}

```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 73   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

```

.input_form .image{
    width: 100%;
    display: flex;
}
.image .selectImg{
    display: none;
}
.image .selectLabel{
    display: block;
    width: 300px;
    height: 50px;
    border: 1px solid #287551;
    border-radius: 6px;
    cursor: pointer;
    line-height: 50px;
    text-align: center;
    font-size: 15px;
    font-family: mon-reg;
}
.image .fileName{
    width: 200px;
    height: 50px;
    display: block;
    color: #000;
    line-height: 50px;
    text-align: center;
    font-size: 15px;
    font-family: mon-reg;
}
.input_form .title, .content{
    border: 1px solid #287551;
    border-radius: 6px;
    font-size: 15px;
    font-family: mon-reg;
    padding-left: 11px;
}
.input_form .title{
    width: 100%;
    height: 40px;
    margin: 20px 0;
}
.input_form .content{
    width: 100%;
    height: 200px;
    resize: none;
    padding-top: 11px;
}
.input_form .btn{
    width: 150px;
    height: 40px;
    border-radius: 6px;
    background: #287551;
    color: #fff;
}

```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 74   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

```

        font-family: mon-reg;
        font-size: 15px;
        margin-top: 15px;
    }
    .addNew .back{
        width: 200px;
        font-size: 15px;
        font-family: mon-reg;
        margin: 20px 20px;
        display: flex;
        align-items: center;
        justify-content: center;
        color: #000;
    }
    .back .arrow{
        width: 18px;
        height: 18px;
        margin-right: 10px;
    }

```

### Файл Admin.js

```

import React from 'react';
import styles from './Admin.module.css';
import axios from 'axios';
import { Link } from 'react-router-dom';
import del from '../../../icons/remove.png'

class Admin extends React.Component{
    state = {
        title: '',
        content: '',
        news: []
    };
    componentDidMount = () => {
        this.getNew();
    };
    getNew = () => {
        axios.get('http://localhost:8080/api')
            .then((responce) => {
                const data = responce.data;
                this.setState({news: data});
                console.log('Data received!');
            })
            .catch(() => {
                alert("Error data!");
            });
    };
    click = () => {
        alert("Статтю видалено");
    }
    displayNew = (news) => {
        if(!news.length) return null;

```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 75   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

```

        return news.map((news, index)=>(
            <div key={index} className={styles.item}>
                <p>{news.title}</p>
                <p>{news.date.substr(0,10)}</p>
                <Link onClick={this.click} to="/del"
className={styles.remove}>
                    <img src={del} alt="Delete" />
                </Link>
            </div>
        ));
    };
    render() {
        console.log("efrgt");
        return(
            <div className={styles.admin}>
                <div className={styles.content}>
                    <div className={styles.head}>
                        <p className={styles.news}>Новини</p>
                        <Link className={styles.create_btn}
to="/admin/add-new">Створити</Link>
                    </div>
                    <div className={styles.titles}>
                        <p>Назва</p>
                        <p>Дата</p>
                    </div>
                    <div className={styles.items}>
                        {this.displayNew(this.state.news)}
                    </div>
                </div>
            </div>
        );
    };
    export default Admin;

```

### Файл **Admin.module.css**

```

.admin{
    width: 100%;
    height: 100vh;
    position: relative;
}
.admin .content{
    width: 60%;
    position: absolute;
    top: 50%;
    left: 50%;
    transform: translate(-50%,-50%);
    border: 1px solid #287551;
    border-radius: 10px;
    padding-bottom: 50px;
}
.content .head{
    width: 100%;
    height: 80px;

```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 76   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

```

        display: flex;
        justify-content: space-between;
        align-items: center;
        padding: 0 50px;
    }
    .head .news{
        font-family: mon-medium;
        font-size: 25px;
    }
    .head .create_btn{
        width: 120px;
        height: 40px;
        background: #287551;
        border-radius: 5px;
        display: flex;
        align-items: center;
        justify-content: center;
        color: #fff;
        font-family: mon-reg;
        font-size: 15px;
    }
    .content .titles{
        width: 90%;
        height: 20px;
        align-items: center;
        font-size: 16px;
        font-family: mon-medium;
        display: grid;
        grid-template-columns: 75% 20% 5%;
        margin: 20px auto;
        padding: 0 0 0 25px;
    }
    .item{
        width: 90%;
        height: 50px;
        border-radius: 5px;
        border: 1px solid #287551;
        align-items: center;
        margin: 10px auto;
        font-size: 16px;
        font-family: mon-reg;
        padding: 0 20px;
        display: grid;
        grid-template-columns: 75% 20% 5%;
    }
    .item input{
        border: 1px solid red;
    }
    .remove img{
        width: 30px;
        height: 30px;}

```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 77   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

## Файл New.js

```
import React from 'react';
import styles from './New.module.css';
import { useParams } from 'react-router-dom'
import Footer from '../../components/Footer/Footer';
import {Link} from 'react-router-dom';
import arrow from '../../icons/arrow.png';
import axios from 'axios';

const GetID=()=>{
  const { id } = useParams()
  console.log("ID:" + id);
  return <p>{id}</p>
}

class New extends React.Component{
  state ={
    title:'',
    content:'',
    source:'',
    news: []
  };
  componentDidMount =()=>{
    this.getNew();
    // const id = this.props.match.params.id;
    // console.log(id);
  };

  getNew =()=>{
    axios.get('http://localhost:8080/api')
      .then((responce) =>{
        const data = responce.data;
        this.setState({news: data});
        console.log('Data received!');
      })
      .catch(()=>{
        alert("Error data!");
      });
  };

  render() {
    const new_inf = this.state.news.find(x => x._id ===
'60c1d5712e74ce05047b8cfb');
    console.log({new_inf});
    return (
      <div className={styles.new}>
        <GetID />
        <Link to="/" className={styles.back}>
          <img className={styles.arrow} src={arrow}
alt="Назад" />
          Повернутися назад</Link>
        <div className={styles.content}>
          <img className={styles.image}>
```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 78   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

```

src={"/images/" + new_inf?.source} alt="Зображення статті" />
      <p
className={styles.title}>{new_inf?.title}</p>
      <p
className={styles.text}>{new_inf?.content}</p>
    </div>
    <Footer/>
  </div>
);
};
}

```

export default New;

### Файл **New.module.css**

```

.new{
  width: 100%;
  height: 50rem;
}
.new .content{
  width: 60%;
  border: 1px solid #287551;
  border-radius: 18px;
  padding: 25px 15px;
  margin: 50px auto;
}
.content .image{
  position: relative;
  width: 80%;
  height: 400px;
  left: 50%;
  transform: translate(-50%);
}
.content .title{
  font-family: mon-medium;
  font-size: 20px;
  padding: 50px 0 0 10px;
}
.content .text{
  font-family: mon-reg;
  font-size: 16px;
  color: #131313;
  line-height: 28px;
  padding: 20px 0;
}
.new .back{
  width: 200px;
  font-size: 15px;
  font-family: mon-reg;
  margin: 20px 20px;
  display: flex;
  align-items: center;
  justify-content: center;
}

```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 79   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |



```

        color: #000;
    }
    .back .arrow{
        width: 18px;
        height: 18px;
        margin-right: 10px;
    }

```

### Файл HomePage.js

```

import React from 'react';
import Header from '../components/Header/Header.js';
import Home from '../components/Home/Home.js';
import News from '../components/News/News.js';
import Entrant from '../components/Entrant/Entrant.js';
import About_Us from '../components/About_Us/About_Us.js';
import Details from '../components/Details/Details.js';
import Footer from '../components/Footer/Footer.js';
const HomePage = () => {
    return(
        <div className="homePage">
            <Header/>
            <Home/>
            <News/>
            <Entrant/>
            <About_Us/>
            <Details/>
            <Footer/>
        </div>
    );
};

```

export default HomePage

Файл App.js

```

import React from 'react'
import './App.css';
import HomePage from './pages/HomePage';
import New from './pages/New/New';
import {Route, Switch} from 'react-router-dom';
import AddNew from './pages/AddNew/AddNew';
import Admin from './pages/Admin/Admin';

```

```

const App = ()=> {
    return (
        <div className="App">
            <Switch>
                <Route exact path="/">
                    <HomePage/>
                </Route>
                <Route path="/new/:id">
                    <New />
                </Route>
                <Route exact path="/admin">
                    <Admin/>
                </Route>
            </Switch>
        </div>
    );
};

```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 80   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

```

        <Route path="/admin/add-new">
          <AddNew />
        </Route>
      </Switch>

    </div>
  );
}

```

```
export default App;
```

### Файл **index.css**

```

/* Montserrat */
@font-face {
  font-family: mon-reg;
  src: url(./fonts/Montserrat-Regular.ttf);
  font-style: normal;
  font-weight: 400;
}
@font-face {
  font-family: mon-medium;
  src: url(./fonts/Montserrat-Medium.ttf);
  font-style: normal;
  font-weight: 500;
}
@font-face {
  font-family: mon-bold;
  src: url(./fonts/Montserrat-Bold.ttf);
  font-style: normal;
  font-weight: 700;
}
/* ---- */

```

### Файл **index.js**

```

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

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

// If you want to start measuring performance in your app,
pass a function
// to log results (for example: reportWebVitals(console.log))
// or send to an analytics endpoint. Learn more:

```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 81   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

```
https://bit.ly/CRA-vitals
reportWebVitals();
```

### Файл new.js

```
const mongoose = require('mongoose');
//Schema
const Schema = mongoose.Schema;
const newSchema = new Schema({
  date: {
    type: Date,
    default: Date.now()
  },
  title: String,
  content: String,
  source: String
});

// Models
const New = mongoose.model('New', newSchema);

module.exports = New;
```

### Файл api.js

```
const express = require('express');
const router = express.Router();

const New = require('../models/new.js');

router.get('/', (req, res)=>{
  New.find({ })
    .then((data)=>{
      console.log('Data', data);
      res.json(data);
    })
    .catch((error)=>{
      console.log('error:', daerrorta);
    });
});

router.post('/save', (req, res)=>{
  console.log("Body: ", req.body);
  const data = req.body;

  const newRec = new New(data);

  newRec.save((error)=>{
    if(error){
      res.status(500).json({ msg:"Sorry, server
problems"});
      return;
    }
    return res.json({
```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 82   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

```

        msg: "We received your data"
      });
    });
  });

router.delete('/new/:id', (req, res) => {
  console.log("delete news")
});

module.exports = router;

```

### Файл **server.js**

```

//mongodb database login:admin pass:admin
const express = require('express');
const mongoose = require('mongoose');
const morgan = require('morgan');
const cors = require('cors');
const path = require('path');
const routes = require('./routes/api');

const app = express();
const PORT = process.env.PORT || 8080;
const MONGODB_URL =
'mongodb+srv://admin:admin@news.ykouq.mongodb.net/News?retryWrites
=true&w=majority'

mongoose.connect(MONGODB_URL, {
  useNewUrlParser: true,
  useUnifiedTopology: true
});

mongoose.connection.on('connected', () => {
  console.log('Mongoose is conected!!');
});

app.use(express.json());
app.use(express.urlencoded({extended:false}));

app.use(cors());

// http request longer
app.use(morgan('tiny'));
app.use('/api', routes);

app.listen(PORT, console.log(`Server is starting at ${PORT}
port`))

```

### Файл **package.json**

```

{
  "name": "school",
  "version": "1.0.0",

```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 83   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |

```

"description": "",
"main": "server.js",
"scripts": {
  "test": "echo \"Error: no test specified\" && exit 1",
  "start": "node server.js"
},
"keywords": [],
"author": "",
"license": "ISC",
"dependencies": {
  "axios": "^0.21.1",
  "concurrently": "^6.2.0",
  "cors": "^2.8.5",
  "express": "^4.17.1",
  "mongoose": "^5.12.13",
  "morgan": "^1.10.0",
  "nodemon": "^2.0.7"
}

```

|     |      |          |         |      |                        |      |
|-----|------|----------|---------|------|------------------------|------|
|     |      |          |         |      | ДП.КН 21.466.11.000 ПЗ | Арк. |
|     |      |          |         |      |                        | 84   |
| Зм. | Арк. | № докум. | Підпис. | Дата |                        |      |