

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ЛЬВІВСЬКИЙ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ІВАНА ФРАНКА**

Факультет прикладної математики та інформатики

(повне найменування назва факультету)

кафедра дискретного аналізу та інтелектуальних систем

(повна назва кафедри)

ДИПЛОМНА РОБОТА

Розробка системи для побудови веб-сайтів

Студента 4 курсу, групи ПМі-45,
спеціальності 122 – комп'ютерні науки

Козяк В.Р.

(підпис)

(прізвище та ініціали)

Керівник

Коркуна Н.М.

(підпис)

(прізвище та ініціали)

Рецензент

(підпис)

(прізвище та ініціали)

Львів 2023

ЗМІСТ

ВСТУП	3
1. ФОРМУЛЮВАННЯ ЗАВДАННЯ ТА ВИБІР МЕТОДОЛОГІЇ ДЛЯ СТВОРЕННЯ ПРОЕКТУ	7
1.1 Постановка задачі	7
1.2 Опис програмних засобів та технологій	9
1.2.1 JavaScript	9
1.2.2 Express Framework	11
1.2.3 React	12
1.2.4 Redux	14
2. ОПИС ІНТЕРФЕЙСУ	15
2.1 Header сторінка	15
2.2 Основна інформація	16
2.3 Про нас	17
2.4 Наші роботи	19
2.5 Бренди, які нас обирають	19
2.6 Footer сторінка	20
2.7 Допоміжний функціонал	22
3. ПРИКЛАД РЕАЛІЗАЦІЇ	23
ВИСНОВКИ	27
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	28

ВСТУП

Сучасний Інтернет є потужною інформаційною системою, яка забезпечує комунікацію між мільйонами користувачів. Широкий доступ до мережі дає можливість переглядати мільйони веб-сайтів, що створені за допомогою мови розмітки HTML та стилів CSS. HTML, що означає Hyper Text Markup Language, є мовою розмітки гіпертексту, яка використовується для створення веб-сторінок, а CSS (Cascading Style Sheets) відповідає за стилізацію та оформлення цих сторінок. Однак, разом з HTML і CSS, сьогодні використовуються інші технології, такі як React та Node.js, для розробки веб-застосунків. React є потужною бібліотекою JavaScript, яка дозволяє створювати складні інтерфейси користувача. Використання React дозволяє розбити веб-сторінку на компоненти, що спрощує розробку, управління станом та підтримку коду. Node.js, з іншого боку, є середовищем виконання JavaScript, яке дозволяє виконувати JavaScript на сервері. Використання Node.js дозволяє розробникам створювати повноцінні веб-застосунки на базі JavaScript, які можуть взаємодіяти з базами даних, здійснювати обробку запитів користувачів та забезпечувати різноманітні функції серверної частини. Створення персонального веб-сайту, використовуючи сучасні технології, не тільки дозволяє розповісти про свій досвід та знання, але і дає змогу обмінюватися корисною інформацією з колегами та здобувачами освіти, а також їхніми батьками. За допомогою React та Node.js можна створити веб-застосунок який виконує широкий спектр завдань, серед яких можна виділити наступні основні:

1. Забезпечення доступу до інформації та ресурсів, які знаходяться в Інтернеті. Веб-застосунок може використовувати HTTP-протокол для взаємодії з серверами та отримання необхідних даних.
2. Взаємодія з користувачами через інтерфейс веб-застосунку, що дозволяє користувачам виконувати різноманітні завдання та отримувати необхідну інформацію. React дозволяє створювати динамічні інтерфейси, які реагують на дії користувача без перезавантаження сторінки.
3. Зберігання та обробка даних, що надходять від користувачів, включаючи персональну інформацію та інформацію про їх дії в системі. Node.js може

використовуватися для збереження даних в базі даних та забезпечення їх обробки.

4. Забезпечення безпеки веб-застосунку та даних, що знаходяться в ньому. Застосування належних заходів безпеки, таких як шифрування даних та аутентифікація користувачів, може бути реалізовано за допомогою відповідних бібліотек та модулів.
5. Підтримка та оновлення веб-застосунку, що дозволяє йому працювати ефективно та забезпечувати необхідні функції для користувачів. Розробка з використанням React і Node.js дозволяє зручно розширювати функціональність та оновлювати веб-застосунок з мінімальними зусиллями.
6. Реклама продукції, послуг, ідей; продаж товарів, послуг, інформації, ідей. Веб-застосунок може включати функціонал для реклами, продажу товарів, послуг, інформації та ідей. За допомогою React та Node.js можна створювати ефективні інтерфейси для рекламних банерів, обробляти платежі, керувати складом товарів і забезпечувати зручний процес покупки для користувачів.
7. Створення платформи для спілкування та взаємодії користувачів з метою обміну думками, ідеями та інформацією. React може використовуватися для створення динамічних форумів, коментарів та соціальних мереж, де користувачі можуть взаємодіяти, обговорювати теми та ділитися своїми думками.
8. Надання послуг онлайн, таких як бронювання, онлайн-курси, консультації та інше. Веб-застосунок на базі React та Node.js може забезпечувати можливість замовлення та оплати послуг онлайн, надавати доступ до електронних курсів та організувати відео-консультації з фахівцями.

Загалом, використання React та Node.js у розробці веб-застосунків відкриває широкі можливості для створення функціональних, ефективних і інтерактивних інтернет-проектів.

Важливість мети створення інтернет-проекту полягає в тому, що вона визначає напрямок і характеристики сайту. Мета створення інтернет-проекту може бути різноманітною залежно від його типу та призначення. Основні цілі, які

можуть стояти перед створенням інтернет-проекту, включають:

- 1) Представлення бізнесу або товарів та послуг компанії в Інтернеті, залучення нових клієнтів та збільшення продажів.
- 2) Створення власного бренду та забезпечення його реклами в Інтернеті.
Розширення сфери впливу та комунікації зі своєю аудиторією, залучення нових користувачів та підтримка взаємодії зі старими клієнтами.
- 3) Публікація та поширення контенту, включаючи новини, статті, фото та відео матеріали.
- 4) Створення платформи для спілкування та взаємодії користувачів з метою обміну думками, ідеями та інформацією.
- 5) Надання послуг онлайн, таких як бронювання, онлайн-курси, консультації та інше.
- 6) Забезпечення доступу до інформації та ресурсів, які можуть бути корисними для користувачів.
- 7) Забезпечення можливості зв'язку з різними інтернет-сервісами та платформами для обміну даними та інтеграції з ними.

Практично 95% сайтів, що з'являються в Інтернеті, мають комерційну спрямованість і створюються з метою забезпечення прибутку для їх власників у майбутньому. Основними завданнями, які може вирішувати якісний інтернет-ресурс, є реклама та продаж власної або партнерської продукції, монетизація сайту за допомогою рекламних або реферальних програм і інших методів.

Некомерційні організації, як правило, створюють свої проекти з благодійною метою, а також для обміну інформацією, наприклад, фото і відеофайлами. Однак, фактично, надання інформації або послуг є засобом залучення відвідувачів до даного ресурсу для здобуття, наприклад, статистичної інформації або показу реклами, якщо цей ресурс є рекламним майданчиком.

З поширенням інтернету все більше людей розуміють, що наявність власного сайту - це необхідна умова успішного онлайн-бізнесу, особливо в умовах пандемії. Вірно розроблений сайт може запропонувати всі необхідні послуги і зберегти кілька відділів, що дозволить знизити витрати на персонал. Проте створення веб-

сайту - це складна задача, яка вимагає залучення професіоналів, а це не завжди дешево. Тому розробка системи для створення та управління веб-сторінками є актуальною, оскільки це дає можливість власникам бізнесу та всім охочим створити свій веб-сайт без додаткових витрат на професійних розробників.

1. ФОРМУЛЮВАННЯ ЗАВДАННЯ ТА ВИБІР МЕТОДОЛОГІЇ ДЛЯ СТВОРЕННЯ ПРОЕКТУ

1.1 Постановка задачі

У результаті розробки має бути створена система для побудови веб-сайтів, що складається з двох сайтів та одного сервера.

Перший сайт призначений для створення та оновлення шаблону веб-сторінки, з можливістю додавання та видалення блоків.

Другий сайт відображатиме контент та блоки, які були обрані на першому сайті.

Сервер буде допомагати взаємодіяти між сайтами та приймати запити на отримання та створення шаблону веб-сторінки. Обидва сайти будуть реалізовані як "single page application", що динамічно підвантажуються у вигляді AJAX та організують взаємодію з користувачем через HTML, CSS та JavaScript.

Програмна реалізація системи для побудови веб-сайтів, що складається з двох сайтів та одного сервера, може мати наступний архітектурний підхід:

1. Серверна частина:

- Node.js для створення серверного додатку.
- Express.js для створення API-інтерфейсу сервера, який буде обробляти запити від сайтів.
- Реалізація різних маршрутів на сервері для обробки запитів.

2. Перший сайт (для створення та оновлення шаблону):

- React для побудови клієнтського інтерфейсу сайту.
- Redux для ефективного управління та оновлення стану додатка. За його допомогою можемо централізувати дані та стан додатка в одному місці, щоб забезпечити зручну та передбачувану роботу зі змінними стану.
- Компоненти, які дозволяють користувачу додавати, видаляти та редагувати блоки шаблону.
- З'єднання з сервером за допомогою API-інтерфейсу для надсилання запитів, пов'язаних зі створенням шаблону.

3. Другий сайт (для відображення контенту та блоків):

- React для побудови клієнтського інтерфейсу сайту.
- Redux для ефективного управління та оновлення стану додатка. За його допомогою можемо централізувати дані та стан додатка в одному місці, щоб забезпечити зручну та передбачувану роботу зі змінними стану.
- Компоненти, які отримуватимуть контент та блоки з сервера і відобразять їх на сторінці.
- З'єднання з сервером за допомогою API-інтерфейсу для отримання контенту та блоків, обраних на першому сайті.

Основна структура системи для побудови веб-сайтів зображена на рисунку 1.1.

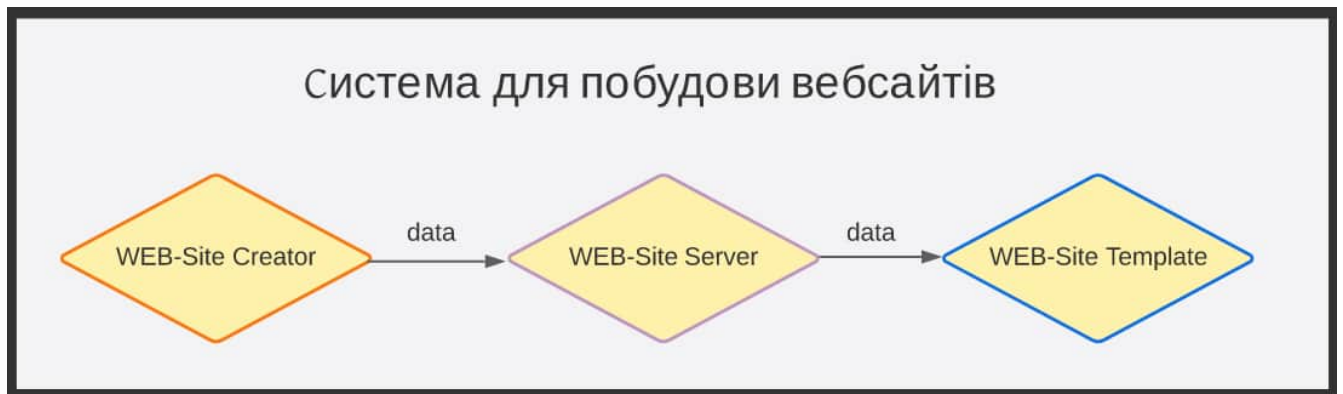


Рис. 1.1 Система для побудови веб-сайтів

1.2 Опис програмних засобів та технологій

Для розробки системи для побудови веб-сайтів, було використано низку технологій та програмних засобів. Основні з них такі:

- a) Мова програмування JavaScript - це мова програмування, що використовується для розробки веб-додатків та додатків на бічному боці.
- b) Node.js - це середовище виконання JavaScript, що дозволяє запускати JavaScript-код на сервері.
- c) Express - це фреймворк для розробки веб-додатків на Node.js. Він надає різноманітні засоби для роботи з маршрутами, запитами та відповідями на сервер.
- d) React - це js-бібліотека для побудови інтерфейсів. Вона дозволяє розробникам створювати динамічні та інтерактивні інтерфейси, які оновлюються без перезавантаження сторінки.
- e) Redux - це js-бібліотека для керування та оновлення стану програми. Вона дозволяє розробникам легко керувати станом додатку та оновлювати його згідно зі змінами, що відбуваються в додатку.

Ці технології та програмні засоби дозволяють розробникам ефективно створювати веб-додатки з високоякісними інтерфейсами та оптимізованою функціональністю. Використання цих засобів може підвищити продуктивність розробки та забезпечити якість виконання проекту.

1.2.1 JavaScript

JavaScript - це скриптова мова програмування, яка використовується для створення динамічних інтерактивних веб-сторінок і додатків. Вона виконується в браузері користувача і дозволяє додавати веб-сторінкам різноманітні ефекти, анімацію, перевірку форм і багато іншого. JavaScript дозволяє змінювати HTML-код веб-сторінки під час її виконання, додавати інтерактивність до елементів сторінки, виконувати асинхронні запити до сервера та інше.

Однією з особливостей JavaScript є те, що вона є мовою з відкритим вихідним кодом, що дозволяє розробникам вносити свої внески у її розвиток і покращення. Також, JavaScript має велику кількість бібліотек і фреймворків, що дозволяє зменшити час на розробку проектів та покращити їхню якість.

Застосування JavaScript у веб-розробці є надзвичайно популярним, оскільки вона дозволяє створювати інтерактивні та багатофункціональні веб-сайти та додатки. Вона використовується в багатьох веб-фреймворках, таких як Angular, React та Vue, що дозволяє створювати складні веб-додатки з великою кількістю функцій та можливостей.

1.2.2 Express Framework

Express - це один з найпопулярніших фреймворків для побудови веб-додатків на Node.js. Він дозволяє розробникам швидко та просто побудувати веб-сервер та обробляти HTTP-запити, такі як GET, POST, PUT і DELETE.

Express є легким та гнучким фреймворком, що дозволяє розробникам використовувати його за різними потребами - від створення малих і простих веб-сайтів до великих та складних веб-додатків. Він не нав'язує структуру проекту, тому розробники можуть використовувати будь-яку організацію файлів та папок, яку вони вважають зручною.

Express також має велику кількість модулів та плагінів, які дозволяють розширювати його функціональність. Наприклад, модуль `body-parser` дозволяє розробникам легко отримувати дані з HTTP-запитів, таких як POST і PUT, а модуль `morgan` дозволяє логувати HTTP-запити для налагодження та моніторингу додатку.

У загальному, Express - це потужний фреймворк, який дозволяє розробникам створювати веб-додатки швидко, ефективно та зручно. Він є відмінним вибором для будь-якого проекту, який потребує побудови веб-інтерфейсу.

1.2.3 React

React - це бібліотека JavaScript, яка використовується для створення інтерфейсів користувача. Вона дозволяє розробникам будувати веб-додатки з високою продуктивністю та ефективністю, а також забезпечує зручний спосіб управління станом компонентів. React використовує підхід "компонентів", який дає можливість розбити веб-додаток на окремі блоки, що спрощує розробку та підтримку коду.

Однією з основних переваг React є можливість використовувати JSX - спеціальний синтаксис, який дозволяє описувати інтерфейс користувача, об'єднуючи HTML та JavaScript. React також забезпечує можливість використовувати "віртуальний DOM", що дозволяє зменшити кількість маніпуляцій з реальним DOM та покращити продуктивність додатка.

За допомогою React можна створювати різні типи веб-додатків, від простих односторінкових додатків до складних масштабних проєктів. React також добре поєднується з іншими технологіями, такими як Redux, TypeScript, Next.js та іншими.

Всі ці функції та можливості роблять React однією з найпопулярніших технологій для розробки веб-додатків на сьогоднішній день. React використовує інкапсульовані компоненти, які керують власним станом з яких будуються складні інтерфейси. Оскільки логіка компонентів написана на JavaScript, замість шаблонів з легкістю можемо передавати складні дані у додатку і зберігати стан окремо від DOM. React також може рендеритись на сервері, використовуючи Node, і приводити в дію мобільні додатки, які використовують React Native.

У React, функціональні компоненти — це спрощений спосіб написання компонентів, що складаються тільки з метода для рендера і не мають власного стану. Замість визначення класу, який поширює `React.Component`, ми можемо створити функцію, яка приймає пропси і повертає те, що треба відрендерити. Функціональні компоненти коротші у написанні, і більшість компонентів можна оформити таким чином.

Компонентно-орієнтований підхід дає можливість легко змінювати наявні

компоненти і використовувати код повторно. У компонентів, які створені під час роботи над тим чи іншим проектом, немає додаткових залежностей. Можливість з легкістю наново використовувати вже наявний код підвищує швидкість розробки та спрощує процес тестування.

1.2.4 Redux

Redux - це відкрита бібліотека для керування станом додатка у JavaScript-додатках. Вона використовується для керування станом додатка в більш складних додатках, які мають багато інтерактивної взаємодії з користувачем.

Redux дозволяє зберігати стан додатка в одному місці - "store" - і змінювати його через "actions". Дані в "store" є незмінними і не можуть бути змінені безпосередньо. Зміна даних можлива тільки через відправлення "actions", які містять відомості про те, які зміни потрібно зробити.

Коли "action" відправлений, він передається до "reducer", який змінює стан додатка, відповідно до вказівок у "action". Редюсер повертає новий стан додатка, який зберігається в "store". Компоненти додатка можуть підписуватися на зміни в "store" і автоматично оновлювати свій стан, коли він змінюється.

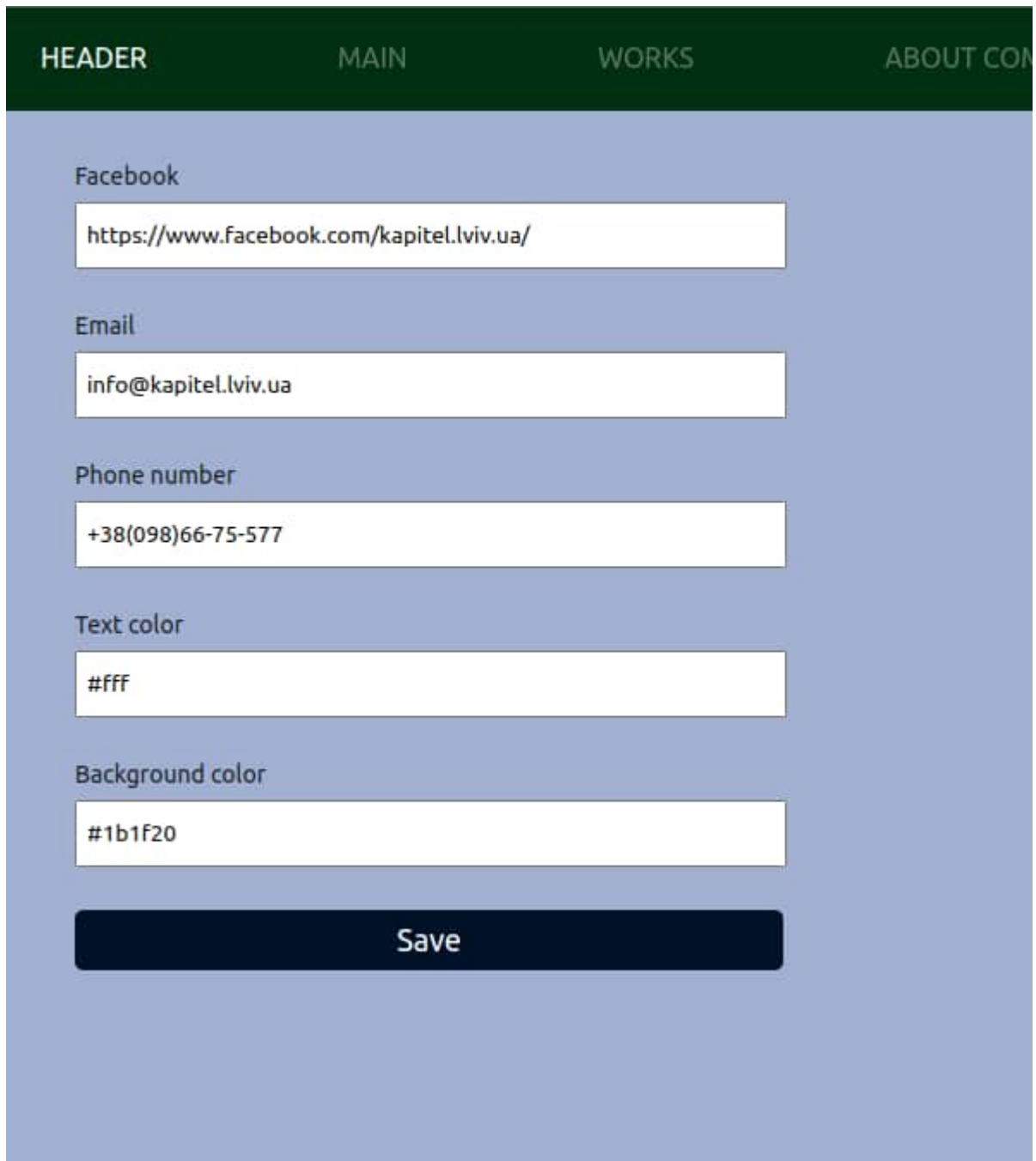
Redux робить управління станом додатка більш прогнозованим і простішим для розуміння, особливо в більш складних додатках з багатьма компонентами та станами. Він також дозволяє зручно реалізувати такі поняття, як "історія" і "перезавантаження сторінки", що допомагає підтримувати зручний та прогнозовуваний додаток.

2. ОПИС ІНТЕРФЕЙСУ

У програмі "WebSite Creator" кожна веб-сторінка має стандартну структуру, яка складається з двох основних елементів: шапки та основної частини. Шапка відповідає за функціонал збереження шаблону сайту. Основна частина містить усю інформацію та функціонал сторінки. Ці компоненти присутні на кожній веб-сторінці програми "WebSite Creator".

2.1 Header сторінка

Рисунок 2.1 зображає сторінку "Header" веб-застосунку. На цій сторінці користувач може налаштувати шапку свого сайту, встановивши такі параметри, як колір тексту та фону, електронну адресу, номер телефону та посилання на соціальні мережі, такі як Facebook.



The image shows a configuration form for a website header. At the top, there is a dark green navigation bar with the text 'HEADER', 'MAIN', 'WORKS', and 'ABOUT COM'. Below this, the form is set against a light blue background. It contains several input fields: 'Facebook' with the URL 'https://www.facebook.com/kapitel.lviv.ua/', 'Email' with 'info@kapitel.lviv.ua', 'Phone number' with '+38(098)66-75-577', 'Text color' with '#fff', and 'Background color' with '#1b1f20'. A dark blue 'Save' button is located at the bottom of the form.

Рис. 2.1 Header сторінка

2.2 Основна інформація

Рисунок 2.2 зображає сторінку "Інформація" веб-застосунку. На цій сторінці користувач може налаштувати основну інформацію свого сайту, встановивши такі параметри, як колір тексту та фону, заголовок, опис та зображення для заднього фону.

HEADER MAIN WORKS ABOUT

Title

KAPITEL

Description

Ремонтно-будівельна компанія у Львові та Івано-Франківську KAPITEL. Ми впевнені у нашій якості, тому прописуємо в договорі гарантію на ремонтні роботи Якщо протягом 5 років виникнуть недоліки в нашому ремонті – то ми відремонтуємо все власним коштом. Ви можете спілкуватися з виконробами в онлайн-режимі за допомогою месенджерів, а також скористатися додатковою послугою, та спостерігати за ремонтом завдяки камерам на об'єкті.

Text color

#fff

Background color

#b8a293

Image

<https://kapitel.lviv.ua/wp-content/uploads/2021/03/Rectangle-1-3>

Save

Рис. 2.2 Сторінка «Основна інформація»

2.3 Про нас

Рисунок 2.3 зображає сторінку "Про нас" веб-застосунку. На цій сторінці користувач може налаштувати інформацію про себе та свій бізнес, встановивши такі параметри, як колір тексту та фону, заголовок, опис та заголовок всього блоку.

HEADER MAIN WORKS ABOUT COMPAN

Main title

Про нас

Title

Ремонтно-будівельна компанія у Львові та Івано-Франківську К

Description

Компанія «Капітель» займається ремонтно-будівельними роботами з 2008 року. За цей період ми реалізували понад 1000 об'єктів під ключ, це приміщення від однокімнатної квартири до відпочинкових комплексів понад 40 000 м.кв. Виконроби та керівництво компанії регулярно відвідують виставки та навчання як закордоном, так і в Україні, з яких привозять нові технології та інструменти які дозволяють пришвидшувати роботу ремонтних бригад та зменшувати людський фактор. В нашому шоурумі вас проконсультують

Text color

#fff

Background color

#1b1f20

Save

Рис.2.3 Сторінка «Про нас»

2.4 Наші роботи

На зображенні представлена сторінка «Наші роботи». Користувач може сконфігурувати своє портфоліо на цій сторінці, вибравши колір тексту та фону, а також назву загального блоку. Крім того, користувач може додавати нові роботи до свого портфоліо, вказуючи посилання на зображення роботи.

Рис. 2.4 Сторінка «Наші роботи»

2.5 Бренди, які нас обирають

На зображенні представлена сторінка "Бренди, які нас обирають" (рис. 2.8), де користувач може налаштувати інформацію про компанії-партнерів, з якими він

співпрацює. Тут можна налаштувати колір тексту та заднього фону, заголовок всього блоку, а також динамічно додавати нових партнерів. Картка містить поле для вводу посилання на логотип компанії, який користувач хоче бачити на своєму сайті.

Рис. 2.5 Сторінка «Бренди, які нас обирають»

2.6 Footer сторінка

На зображенні показана сторінка, яка містить інформацію про низ (“підвал”) веб-сайту. Користувач може відредагувати колір тексту, електронну адресу, номер телефону, години роботи та адресу, за якою його можна знайти.

Email

Phone number

Address

Start of the working day

End of working day

Text color

Background color

Save

Рис 2.6 Footer сторінка

2.7 Допоміжний функціонал

На зображенні показані дві кнопки допоміжного функціоналу. Перша кнопка, призначена для заповнення всіх полів дефолтним контентом, розроблена з метою швидкої демонстрації роботи програми. Ця кнопка автоматично заповнює всі відповідні поля на основі попередньо заданого контенту. Ця функція дозволяє нам швидко перевірити, як програма працює з реальними даними без необхідності вводити їх вручну кожного разу. Це особливо корисно під час презентацій або тестування програми, коли вам потрібно продемонструвати її функціонал швидко і ефективно. Друга кнопка призначена для створення шаблону для потрібного нам сайту на основі всіх наших полів. Це означає, що програма збирає всі введені дані з полів, які ви заповнили, і створює шаблон, який може бути використаний для створення веб-сайту з використанням цих даних.

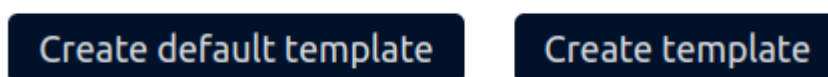


Рис 2.7 Footer сторінка

3. ПРИКЛАД РЕАЛІЗАЦІЇ

Для створення шаблону сайту необхідно заповнити поля сторінок веб-сервісу «Website Creator» і натиснути на кнопку «Створити». Після цього програма формує js-об'єкт на основі введених даних та відправляє його на сервер. На малюнках 3.1, 3.2 і 3.3 зображено отриманий об'єкт.

```

1  export const CONTENT = {
2    header: {
3      work_email: "info@kapitel.lviv.ua",
4      work_phone: "+38(098)66-75-577",
5      facebook_account: "https://www.facebook.com/kapitel.lviv.ua/",
6      header_color: "#fff",
7      color_background: "#1b1f28",
8    },
9    main: {
10     background_image:
11       "https://kapitel.lviv.ua/wp-content/uploads/2021/03/Rectangle-1-3-scaled.jpg",
12     main_title: "KAPITEL",
13     main_color: "#fff",
14     background_color: "#b8a293",
15     main_description:
16       "Ремонтно-будівельна компанія у Львові та Івано-Франківську KAPITEL. Ми впевнені у нашій якості, тому прописуємо в договорі гарантію на ремонтні роботи Якщо
17   },
18   work: {
19     background_color: "#fff",
20     header_title: "Портфоліо",
21     work_color: "#000",
22     data: [
23       {
24         id: 0,
25         img: "https://kapitel.lviv.ua/wp-content/uploads/2022/09/94f8b3f3-4aab-4338-999e-f765ef8b776a.jpg",
26       },
27       {
28         id: 1,
29         img: "https://kapitel.lviv.ua/wp-content/uploads/2023/01/5ea1b2c9-7d8c-421d-af3d-f2c4b25253bc.jpg",
30       },
31       {
32         id: 2,
33         img: "https://kapitel.lviv.ua/wp-content/uploads/2023/01/19bc6ad7-d66d-43cb-91cf-9cb7d20598da.jpg",
34       },

```

Рис. 3.1 Згенерований об'єкт

```

34     },
35     {
36       id: 3,
37       img: "https://kapitel.lviv.ua/wp-content/uploads/2023/01/7eed2774-bdc8-4e05-8b2e-4e13878ef656.jpg",
38     },
39     {
40       id: 4,
41       img: "https://kapitel.lviv.ua/wp-content/uploads/2023/01/6ad40392-e7b4-4481-9ae3-4b1a286858d1.jpg",
42     },
43     {
44       id: 5,
45       img: "https://kapitel.lviv.ua/wp-content/uploads/2022/07/xphoto_2022-07-04_15-52-14-4.jpg.pagespeed.ic.j-3tAbLN62.webp",
46     },
47     {
48       id: 6,
49       img: "https://kapitel.lviv.ua/wp-content/uploads/2022/07/xphoto_2022-07-04_15-52-15-2.jpg.pagespeed.ic.FMST6i_j-B.webp",
50     },
51     {
52       id: 7,
53       img: "https://kapitel.lviv.ua/wp-content/uploads/2021/10/Greenville9-1.jpg",
54     },
55     {
56       id: 8,
57       img: "https://kapitel.lviv.ua/wp-content/uploads/2021/10/Greenville10-1.jpg",
58     },
59     {
60       id: 9,
61       img: "https://kapitel.lviv.ua/wp-content/uploads/2021/05/Greenville12.jpg",
62     },
63     {
64       id: 10,
65       img: "https://kapitel.lviv.ua/wp-content/uploads/2021/10/zubra4-min1.jpg",
66     },
67     {
68       id: 11,
69       img: "https://kapitel.lviv.ua/wp-content/uploads/2021/10/xzubra3-min1.jpg.pagespeed.ic.K0jVhRm6sp.webp",

```

Рис. 3.2 Згенерований об'єкт (продовження)

```

69 |   img: "https://kapitel.lviv.ua/wp-content/uploads/2021/10/xzubra3-mini.jpg.pagespeed.ic.KQjVhRm6sp.webp",
70 | },
71 | {
72 |   id: 12,
73 |   img: "https://kapitel.lviv.ua/wp-content/uploads/2021/05/xzubra7-min.jpg.pagespeed.ic.XwH-jZE2M.webp",
74 | },
75 | ],
76 | },
77 | brand: {
78 |   background_color: "#fff",
79 |   header_title: "Бренди, які нас обирають",
80 |   brand_color: "#000",
81 |   data: {
82 |     {
83 |       id: 0,
84 |       img: "https://ski.emilyresort.com.ua/wp-content/uploads/2022/11/EMILY-logotypes-all-small-02.png",
85 |     },
86 |     {
87 |       id: 1,
88 |       img: "https://img.lunstatic.net/building-logo-original/194.png",
89 |     },
90 |     {
91 |       id: 2,
92 |       img: "https://avalon-inc.com.ua/img/logo-black.png",
93 |     },
94 |     {
95 |       id: 3,
96 |       img: "https://static.riell.ua/news_cover/large/cuyDH6grzM1oEpXXdmqHld0ax2BoaYtU5dd8e2g6.jpeg",
97 |     },
98 |     {
99 |       id: 4,
100 |      img: "https://www.kandevelopment.com/upload/iblock/efe/TH.png",
101 |     },
102 |     {
103 |       id: 5,
104 |       img: "https://scontent.flwo6-1.fna.fbcdn.net/v/t39.30808-6/309366124_121765280656450_2339628029917582033_n.jpg?nc_cat=106&ccb=1-7&nc_sid=89cbe&nc_ohc=

```

Рис. 3.3 Згенерований об'єкт (продовження)

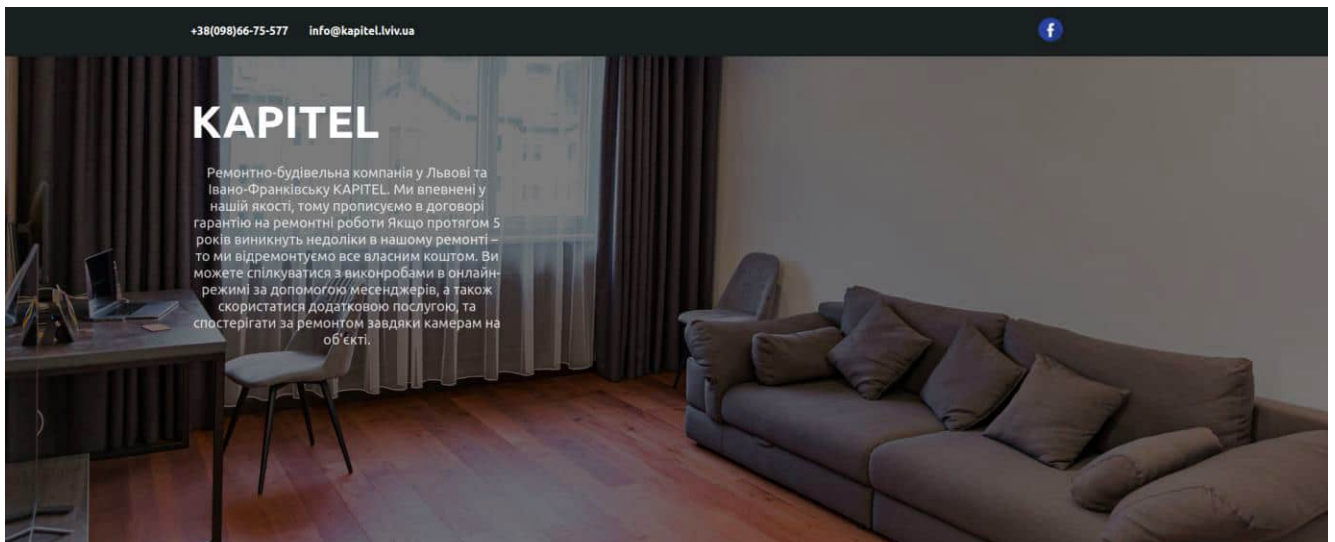
```

104 |       img: "https://scontent.flwo6-1.fna.fbcdn.net/v/t39.30808-6/309366124_121765280656450_2339628029917582033_n.jpg?nc_cat=106&ccb=1-7&nc_sid=89cbe&nc_ohc=
105 |     },
106 |   ],
107 | },
108 | about: {
109 |   header_title: "Про нас",
110 |   title: "Ремонтно-будівельна компанія у Львові та Івано-Франківську KAPITEL",
111 |   about_color: "#fff",
112 |   background_color: "#1b1f20",
113 |   description:
114 |     "Компанія «Капітель» займається ремонтно-будівельними роботами з 2008 року. За цей період ми реалізували понад 1800 об'єктів під ключ, це приміщення від одні
115 |   },
116 | footer: {
117 |   email: "info@kapitel.lviv.ua",
118 |   phone: "+38(098)66-75-577",
119 |   address: "Львів, вул. Івана Пулюя, 40 Івано-Франківськ, вул. І.Івасика, 21",
120 |   footer_color: "#fff",
121 |   background_color: "#1b1f20",
122 |   from_work: "9.00",
123 |   to_work: "18.00",
124 | },
125 | };
126 |

```

Рис. 3.4 Згенерований об'єкт (продовження)

Після того, як менеджер створив js-об'єкт, дані надходять на сервер і використовуються для побудови сайту в веб-застосунку. На рисунках 3.5, 3.6 та 3.7 можна побачити приклад сайту, який був побудований на основі згенерованого об'єкту, що був зображений на рисунках 3.1, 3.2, 3.3 та 3.4.



Портфоліо

Рис. 3.5 Приклад сайту

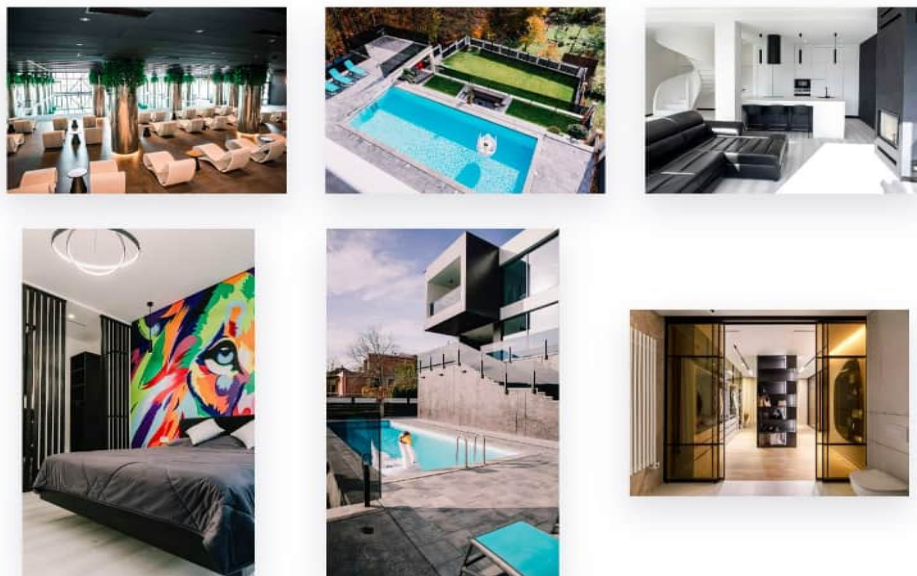


Рис. 3.6 Приклад сайту(продовження)

Про нас

Ремонтно-будівельна компанія у Львові та Івано-Франківську KAPITEL

Компанія «Капітель» займається ремонтно-будівельними роботами з 2008 року. За цей період ми реалізували понад 1000 об'єктів під ключ, це приміщення від однокімнатної квартири до відпочинкових комплексів понад 40 000 м.кв. Виконавці та керівництво компанії регулярно відвідують виставки та навчання як закордоном, так і в Україні, з яких привозять нові технології та інструменти які дозволяють пришвидшувати роботу ремонтних бригад та зменшувати людський фактор. В нашому шоурумі вас проконсультують та допоможуть підібрати все для ремонту - від чорнових матеріалів до меблів, кухонь, підлоги, плитки та іншого. Ми закупимо матеріали та зекономимо ваш час. В процесі ремонту ви будете спілкуватися з персональним виконробом. Він буде займатися організацією всього процесу робіт, вирішувати всі питання, які виникають, та звітувати про виконання роботи. Вам не прийдеться спілкуватися з окремими майстрами.

Бренди, які нас обирають



Рис. 3.5 Приклад сайту(продовження)



Контактна інформація

Адреса: Львів, вул. Івана Пулюя, 40 Івано-Франківськ, вул. В.Івасюка, 21г Телефон: +38(098)66-75-577

Працюємо: з 9.00 до 18.00 Email: info@kapitel.lviv.ua

Рис. 3.7 Приклад сайту(продовження)

ВИСНОВКИ

Результатом даної дипломної роботи є система для побудови веб-сайтів, що дозволяє створювати та редагувати веб-сайти. Користувачі можуть з легкістю створювати свої власні сайти для бізнесу чи особистих потреб за допомогою зручного інтерфейсу веб-застосунку, який відповідає сучасним вимогам і є простим у використанні. Користувачі можуть обрати шаблон або почати з чистого аркуша і створити унікальний дизайн свого веб-сайту. Система надає доступ до різноманітних інструментів та опцій, таких як вибір кольорової палітри, налаштування фону, що дозволяє користувачам створювати візуально привабливі та індивідуальні веб-сайти. Для розробки використовувалися мова програмування JavaScript, фреймворк Express та інші технології, що дозволили детально вивчити всі аспекти реалізації веб-застосунку. Перевірка працездатності була проведена на різних браузерах, включаючи Google Chrome, Safari та Mozilla Firefox, і продемонструвала повну сумісність інтерфейсу в різних браузерах.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Redux – 2019. – Redux Essentials Redux Overview and Concepts: URL: <https://redux.js.org/tutorials/essentials/part-1-overview-concepts> (date of access: 17.05.2023).
2. Express - Node.js web application framework. Express - Node.js web application framework. URL: <https://expressjs.com/> (date of access: 17.04.2023).
3. Ant Design - The world's second most popular React UI framework. Ant Design - The world's second most popular React UI framework. URL: <https://ant.design/> (date of access: 17.05.2023).
4. React – JavaScript-бібліотека для створення користувацьких інтерфейсів. React – JavaScript-бібліотека для створення користувацьких інтерфейсів. URL: <https://uk.reactjs.org/> (дата звернення: 19.03.2023). Redux. Режим доступу: <https://redux.js.org/>.
5. Node.js. Node.js. URL: <https://nodejs.org/uk/> (date of access: 17.05.2023).