Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Інститут комп’ютерних наук та інформаційних технологій

Кафедра автоматизованих систем управління



**Звіт**

до лабораторної роботи №3

з дисципліни

**«Вебтехнології та розробка вебзастосувань»**

На тему: **«Фронтенд-фреймворки»**

Виконав: студент ОІ-11сп

**Данюк Д.О.**

Прийняв: асистент каф. АСУ

**Казарян А.Г.**

Львів –2026

Лабораторна робота №3

**Варіант №5**

**Тема:** Фронтенд-фреймворки

**Мета роботи:** Отримати практичний досвід роботи з одним із сучасних фронтенд-фреймворків. Навчитись розробляти веб-сторінки за допомогою компонентів, керувати станами компонент та працювати з маршрутизацію для переходу між різними відображеннями (сторінками).

**Хід роботи**

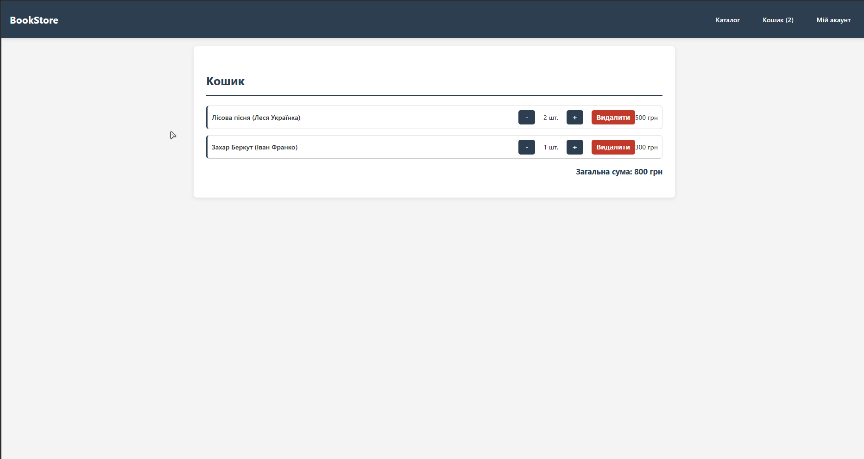
Веб-сайт онлайн-магазину книг

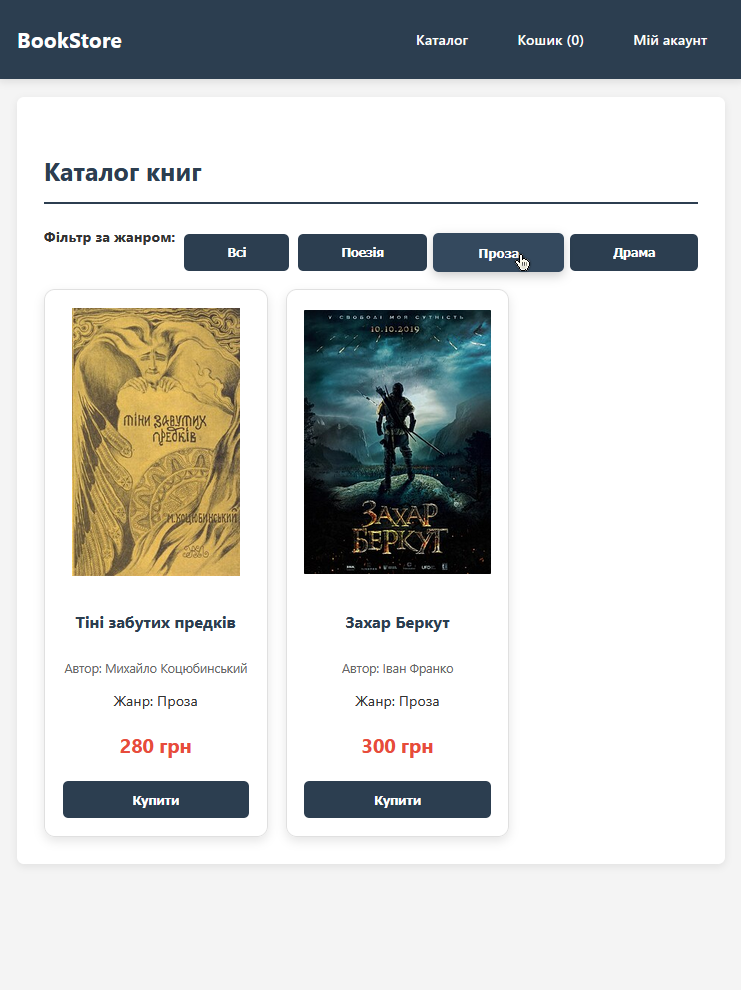
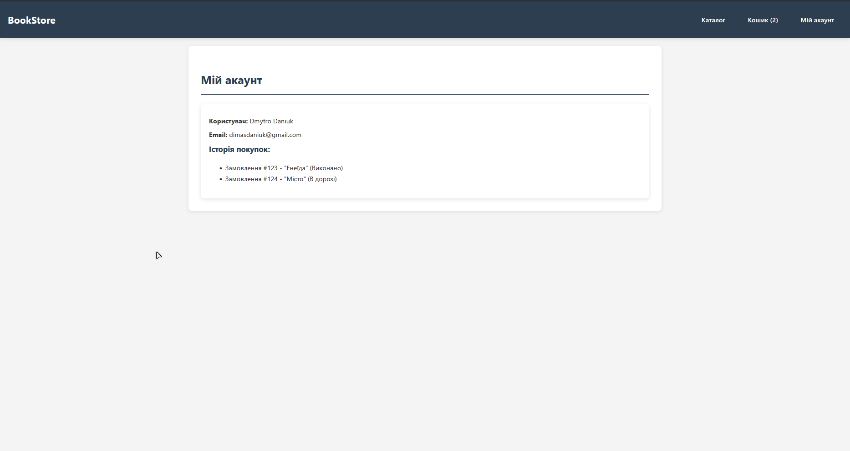
1. Створити компоненти для картки книги, елемента кошика та блоку інформації користувача.

2. У компоненті "Каталог" додати кнопку для фільтрації книг за жанром або автором.

3. Реалізувати маршрутизацію між сторінками без перезавантаження сторінки браузера.

**Посилання на гіт репозиторій:** https://github.com/Ntaviouk/Ntaviouk.github.io

**Результат виконання роботи:**



**Висновок:** Під час виконання лабораторної роботи було отримано практичний досвід роботи з React та закріплено навички створення SPA-застосунків. У процесі розробки онлайн-магазину книг опановано компонентну архітектуру, керування станом і клієнтську маршрутизацію без перезавантаження сторінки. У результаті створено інтерактивний, зручний і масштабований застосунок, що відповідає сучасним вимогам веброзробки.

**Відповіді на контрольні запитання:**

1. **Що таке компонентно-орієнтована архітектура у фронтенд-фреймворках, і які її переваги?** Це підхід, при якому інтерфейс розбивається на незалежні, ізольовані блоки — компоненти (наприклад, кнопка, шапка сайту, картка товару). Переваги: код легше читати та тестувати, компоненти можна повторно використовувати в різних місцях додатку, що зменшує дублювання коду.
2. **Які ключові відмінності між Angular, React та Vue у способі керування станом компонента?** В Angular стан керується в межах класу компонента за допомогою TypeScript. У React для цього використовуються спеціальні функції — хуки (наприклад, useState). У Vue стан визначається всередині функції data().
3. **Що таке Virtual DOM і як він покращує продуктивність у React і Vue?** Virtual DOM — це легка копія реального DOM (дерева HTML елементів), яка зберігається в пам'яті. Коли дані змінюються, фреймворк спочатку оновлює Virtual DOM, порівнює його з попередньою версією, і точково оновлює лише ті елементи на реальній сторінці, які дійсно змінилися. Це значно швидше, ніж повне перезавантаження дерева.
4. **Як працює маршрутизація у фронтенд-фреймворках, і які бібліотеки використовуються для неї у Angular, React та Vue?** Маршрутизація дозволяє перемикати сторінки (компоненти) шляхом зміни URL-адреси без відправлення запиту на сервер і перезавантаження браузера (Single Page Application). Бібліотеки: RouterModule (Angular), react-router-dom (React), vue-router (Vue).
5. **Що таке директиви у Angular та Vue, і як вони відрізняються від умовного рендерингу у React?** Директиви (наприклад, \*ngIf в Angular або v-if у Vue) — це спеціальні атрибути в HTML-шаблоні, які керують відображенням або поведінкою елементівв. У React директив немає; натомість використовується звичайний JavaScript (тернарні оператори або логічне &&) безпосередньо всередині JSX-коду для умовного рендерингу.