

Лабораторна робота №1

Тема: Розгортання проекту. Створення сторінок. Стилiзація

Мета роботи: навчитися розгортати проект; створювати сторінки; стилізувати компоненти.

Практичні завдання

Завдання 1: Розгорніть проект

1. Створіть публічний git репозиторій для проекту на GitHub. Бажано створити повністю пустий репозиторій, щоб було зручніше пушити стартовий NextJS проект.
2. Склонуйте створений репозиторій на локальну машину.
3. Перевірте наявність і версію **NodeJS >= 18** та **npm**
4. Розгорніть проект за допомогою команди **npx create-next-app**
 - a. Ваш проект має використовувати Typescript, eslint, npm, AppRouter, TailwindCSS;
 - b. подумайте, яку структуру проекту Ви хочете отримати і чи потрібна Вам директорія src.

Завдання 2: Створіть сторінки

1. Створіть сторінки з наступними pathname:
 - a. /
 - b. /articles
 - c. /articles/favorite
 - d. /articles/create
 - e. /profile/settings
 - f. /profile/security
2. Створіть окремий layout для всіх сторінок крім /.
3. В створеному layout додайте навігаційне меню для сторінок /articles, /profile/settings, /profile/security.
4. Створіть окремий layout для сторінок /articles*.
5. В ньому додайте навігаційне меню для /articles/favorite і /articles/create.
6. Додайте функціонал підсвічування активного розділу для кожного з меню.

Завдання 3: Отримання даних.

1. На сторінці `/articles` зробіть запит `GET https://jsonplaceholder.typicode.com/posts` і виведіть якусь інформацію з отриманих даних.
2. Покажіть сторінку завантаження поки завантажується сторінка статей.
3. На сторінці `/articles/favorite` зробіть запит `GET https://jsonplaceholder.typicode.com/posts/{ID}`, з будь-якими трьома ID і виведіть якусь інформацію про три отримані статті. Створіть окремий компонент `FavoriteArticle`, який буде показувати цю інформацію.
4. Покажіть окремі, незалежні один від одного індикатори завантаження для кожної з `FavoriteArticle`.
5. Додайте нову сторінку `/articles/{ID}`. При заході на цю сторінку буде йти відповідний запит `https://jsonplaceholder.typicode.com/posts/{ID}` а також `https://jsonplaceholder.typicode.com/posts/1/comment` s. Відобразіть отриману інфу на сторінці.
6. Перетворіть сторінку `/articles/{ID}` на статично генеровану при білді з генерацією статей з ID 1-10.
7. Збілдіть аппку динамічно і запустіть на `next`.
8. Збілдіть аппку статично і запустіть за допомогою npm пакету `http-server`.

Завдання 4: Стилізація.

1. Підключіть SCSS. Перетворіть глобальні стилі на SCSS файл. Створіть декілька глобальних SCSS файлів для змінних, типографії, загального лейауту, міксінів тощо.
2. Створіть модульний SCSS файл для стилізації Вашого меню.
3. Кастомізуйте конфігурацію TailwindCSS, створивши свої брейкпоїнти для розширень екранів, а також свою систему кольорів. Використайте кастомні TailwindCSS класи.

4. Підключіть будь-яку бібліотеку компонентів React (React Bootstrap, Semantic UI React, MUI, Chakra UI, Ant Design).
5. Використайте компонент бібліотеки на своїй сторінці.