

ОПТИМІЗАЦІЯ ПРОДУКТИВНОСТІ ТА SEO ПРИ МАСШТАБУВАННІ ПРОЄКТІВ НА ОСНОВІ REACT І NEXT.JS

UDC 621.855

L. P. Dmytrotsa, Ph.D., Assoc. Prof.; O. T. Starytskyi

OPTIMIZING PERFORMANCE AND SEO WHEN SCALING PROJECTS BASED ON REACT AND NEXT.JS

Оптимізація продуктивності та SEO відіграють ключову роль у сучасній розробці веб-додатків. В умовах зростаючої конкуренції в онлайн-середовищі та підвищених очікувань користувачів щодо якості веб-ресурсів, ці аспекти стають особливо важливими.

Метою роботи було дослідити доступні підходи до оптимізації продуктивності та SEO в проєктах, створених на основі React і Next.js, а також оцінити їхню ефективність і практичну користь для розробників.

Для аналізу методів було підібрано три види рендеру, які по різному впливають на оптимізацію сайту. Було визначено їх основні функції, переваги та недоліки (таблиця 1).

Таблиця 1. Аналіз доступних рендерів

Характеристики	SSR	SSG	CSR
Швидкість завантаження	Помірна	Найвища	Залежить від JS
SEO	Висока	Висока	Низька (без оптимізації)
Динамічність контенту	Найвища	Низька (статичний)	Найвища
Час до відображення	Середній	Найнижчий	Найвищий
Складність реалізації	Середня	Низька	Найнижча

Провівши порівняння та ознайомившись із функціоналом кожного з рендерів, ми обрали SSR, оскільки цей підхід дозволяє ефективно вирішувати задачі SEO та працювати з динамічним контентом. Він забезпечує генерацію готового HTML на сервері для кожного запиту, що покращує індексацію сторінок і створює кращий користувацький досвід завдяки швидкому завантаженню основного контенту. SSR також надає гнучкість у реалізації, що робить його оптимальним вибором для проєктів із динамічними даними.

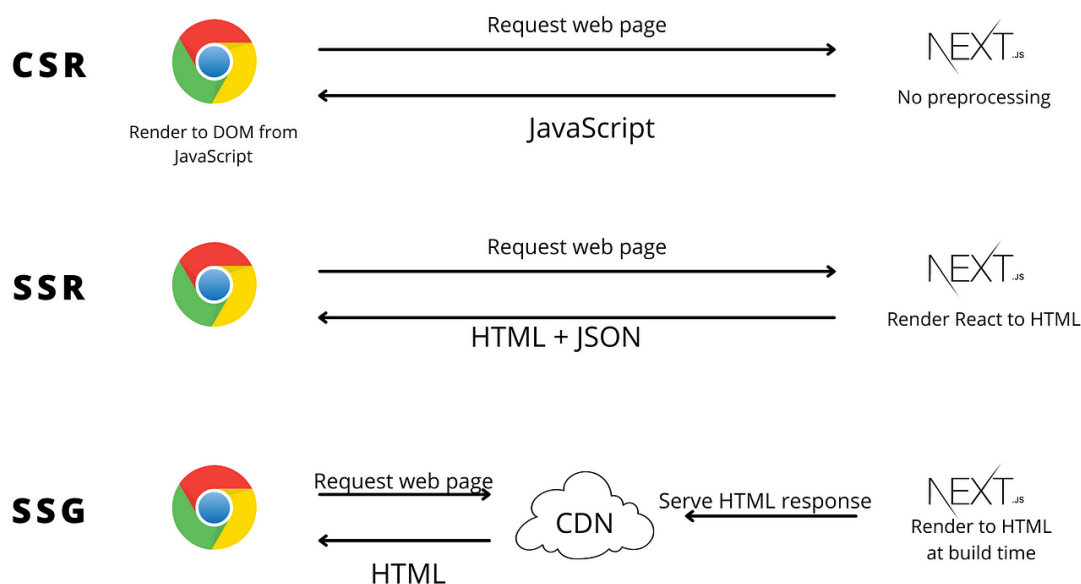


Рисунок 1. Як працюють види рендеру

Висновок: досліджено наявні підходи до рендеру та обрано оптимальний варіант для нашої роботи. На основі вибраної стратегії було обрано основний напрямок для оптимізації. Обраний рендер допоможе нам покращити динамічність та SEO та автоматизувати роботу в певних аспектах коду.

Література

1. React Performance Optimization Techniques // Режим доступу: <https://supertokens.com/blog/5-tips-for-optimizing-your-react-apps-performance>. [дата звернення 12.11.2024].
2. Next.js SEO Optimization: Practical Techniques for Better Ranking. // Режим доступу: <https://nextjs.org/learn-pages-router/seo/introduction-to-seo>. [дата звернення 12.11.2024].
3. Server-Side Rendering and Static Site Generation in Next.js. // Режим доступу: <https://nextjs.org/docs/pages/building-your-application/rendering/static-site-generation> [дата звернення 12.11.2024].
4. Using Google Analytics and Search Console with Next.js. // Режим доступу: <https://nextjs.org/docs/messages/next-script-for-ga>. [дата звернення 12.11.2024].
5. Enhancing Mobile Responsiveness for Better SEO in Next.js. // Режим доступу: <https://nextjs.org/learn-pages-router/seo/web-performance/seo-impact> [дата звернення 12.11.2024].