

ОСОБЛИВОСТІ ОПТИМІЗАЦІЇ ВЕБ-ДОДАТКІВ ПРИ СТВОРЕННІ ЕФЕКТИВНОГО РЕСУРСУ

Партас В.К., к.т.н.¹, Олещук В.І., магістрант²

^{1,2} Національний університет кораблебудування імені адмірала Макарова
^{1,2} Україна, Миколаїв.

¹ viktor.partas@nuos.edu.ua; ² shatalkin1996@gmail.com

Анотація. У статті розглядаються проблеми оптимізації веб-додатків, а також сучасні методи, підходи щодо усунення проблем зниження продуктивності додатків. Приведений аналіз сучасних засобів та інструментів для побудови веб-додатків та їх переваги.

Ключові слова: Оптимізація, веб-додаток, підвищення продуктивності, React, фреймворк.

Метою дослідження є виявлення основних проблем при створенні та оптимізації веб-додатків, а також методи щодо їх усунення. Виявлення оптимальних підходів підвищення продуктивності веб-додатків для підвищення шансів додатку на попадання у топ видачі та підвищення конверсії.

З кожним роком веб-додатки становляться все більшими за розміром, за наповненням контентом та складніші при взаємодії. Розміри файлів, які завантажуються з веб-ресурсів, при цьому збільшуються багатократно, а час очікування користувачів не зменшується. Дослідження більш ніж 160 організацій показує, що кожна секунда затримки призводить до втрати 7% конверсії, 11% перегляду сторінок, задовільність клієнтів знижується на 16%.[1] Тому, підвищення продуктивності веб-додатків є в край актуальною та може бути досягнуто за рішенням наступних задач:

- досягнення мінімально можливого часу завантаження веб-додатку;
- забезпечення мінімально можливого часу з моменту запиту сторінки до моменту появи у користувача можливості продивлятися сторінку та взаємодіяти з нею;
- забезпечення мінімально можливого часу на оновлення інтерфейсу при взаємодії з ним.

Мінімізація часу завантаження, безумовно, є найбільш важливим фактором, який завжди входить до критеріїв оцінки продуктивності будь-якого інтернет-проекту [2]. Тому варто звернути увагу на його складові, до яких можна віднести наступні:

- час на перенаправлення завантаження з одної сторінки на іншу (час на редірект);
- час відповіді серверу;
- час до першого байту;
- час завантаження контенту;
- час повного завантаження сторінки.

Як наслідок, для підвищення продуктивності веб-додатку потрібно виконувати наступні дії:

1. При розробці веб-додатку необхідно зводити до мінімуму кількість редіректів та створювати веб-додатки не більш ніж з одним редіректом с заданого URL на кінцеву сторінку.
2. Оптимізувати роботу с базою даних. Кожен запит займає час, тому, чим більше запитів, тим довший загальний час генерації сторінки. До найбільш поширених прийомів для цього прийнято: очистити базу даних від “сміття”, об’єднати усі стилі в один файл, зміна серверу бази даних на більш продуктивний.
3. Час до першого байту – час необхідний серверу на опрацювання запиту та відправлення його браузеру. На покращення цього показника впливають: оптимізація коду

сторінки, зміна налаштування серверу, оптимізація запитів до бази даних, реалізація каширування.

4. Для скорочення часу завантаження контенту необхідно проводити оптимізацію зображень, шрифтів, CSS та JS, виконати мініфікацію файлів та конкатенацію для зменшення кількості запитів. Застосування методики LazyLoading корисно для «підзавантаження» зображень в необхідний момент під час прокрутки (скролу) сторінки. Залежно від ситуації можна відправляти користувачу лише необхідні йому у даний момент файли, розділяючи код на модулі. Під час повного завантаження сторінки необхідно завчасно моделювати порядок завантаження файлів – “стилі - скрипти”, або скористуватися спеціальними атрибутами (async та defer) для підключення бібліотек.

Веб-додаток може запускатися в різних умовах, тому потрібно проводити тестування продуктивності, проводячи симуляцію різноманітних умов. Наприклад, імітувати одночасне використання веб-додатку великою кількістю людей у тестовому середовищі, щоб виявити як поведе себе веб-додаток, втрачає у швидкості чи ні. Це краще робити у процесі розробки, щоб своєчасно виявляти «баги», збирати дані та коригувати веб-додаток, поки він не буде відповідати необхідним критеріям. Слід створювати різноманітні умови, щоб перевіряти роботу веб-додатку у несприятливих умовах та правити додаток, доводячи його до ідеального стану.

Однією із головних проблем при розробці веб-додатків є синхронізація користувацького інтерфейсу і стану веб-додатку. При кожному оновленні стану веб-додатку потрібно прикласти значних зусиль для оновлення інтерфейсу, залежного від коду. Це дуже неприйнятно для масштабованості та підтримки, так як потрібно порівнювати великі обсяги даних та легко допустити помилку. Для уникнення цього широко використовують фреймворки[3].

Сучасним недоліком розробки веб-додатків за допомогою фреймворків є бездумне використання ресурсів, різноманітних бібліотек та плагінів, нерозуміння компонентного підходу, що призводить до погіршення якості майбутнього продукту. Дослідження показали, що найкращим вибором буде використання React за рахунок наступних переваг:

1. Підвищення продуктивності за рахунок ізоморфного підходу, що дозволяє швидше рендерити сторінки та уникнути дублювання коду.
2. Використання технології Virtual DOM (DocumentObjectModel), що, в свою чергу, також підвищує продуктивність навантажених веб-додатків, за рахунок того що DOM не оновлюється кожен раз після змін.
3. Завдяки методів повторного використання коду, можна застосовувати єдиний код веб версії і для створення мобільного додатку.

Висновок. На даний момент існує чітка залежність між продуктивністю веб-додатку та його позиціями у пошуковій видачі. Тому для покращення шансів попадання веб-додатку у топ - пошук, підвищення конверсії та потоку відвідувачів необхідно приділяти достатньо уваги підвищенню продуктивності веб-додатку.

REFERENCES

- [1] Ilya Grigorik (2014). Пример веб-производительности. Блог компании Infobox. Отримано з <https://habr.com/ru/company/infobox/blog/226289/>.
- [2] SpySerpTeam (2017). Удваиваем скорость загрузки сайта с помощью 7 продвинутых рекомендаций. Отримано з <https://spyserp.com/ru/blog/double-up-website-speed>.
- [3] Ekaterina Novoseltseva (2017). Web performance optimization techniques. Отримано з <https://apiumhub.com/tech-blog-barcelona/web-performance-optimization-techniques/>.

Viktor Partas, Vladyslav Olieshchuk

Features of web application optimization in creating an effective resource

Annotation. *The article discusses the problems of optimizing web applications, as well, as modern methods, approaches to eliminate the problems of decreasing application performance. Analysis of modern tools and tools for building web applications and their benefits.*

Keywords: *Optimization, web application, performance increasing, React, framework.*