

СРАВНИТЕЛЬНЫЙ АНАЛИЗ СКОРОСТИ РЕНДЕРИНГА ПРИЛОЖЕНИЙ НА REACT JS И VUE JS

Морщина Л.В., Столбова Е.Д.

(Университет ИТМО)

Научный руководитель – доцент, кандидат педагогических наук, Государев И.Б.

(Университет ИТМО)

Аннотация. В ходе данной работы проведен эксперимент производительности приложений, написанных с помощью фреймворка Vue JS и библиотеки React JS (далее - фреймворки). Определены критерии эффективности рендеринга, на основании которых проведен сравнительный анализ. В качестве предмета сравнения для каждого из исследуемых фреймворков разработаны базовые приложения ToDo List.

Введение. В настоящее время технологии веб-разработки являются неотъемлемой частью современного мира. Бизнес, организации, государственные компании и корпорации используют веб-приложения или веб-сайты. Язык JavaScript является основополагающим для разработки frontend-части приложения: реализации удобных и отзывчивых пользовательских интерфейсов различной сложности. В связи с этим, технологии веб-разработки стремительно развиваются, стараясь облегчить разработку нового функционала. Использование фреймворков и готовых библиотек считается уже неотъемлемой частью процесса разработки веб-приложения.

Основная часть. Выбор фреймворка (библиотеки) для использования является сложным и нетривиальным вопросом, так как при выборе следует опираться на достоинства и недостатки каждого. Также немаловажную роль играет сложность и архитектура разрабатываемого приложения.

В качестве фреймворков для анализа были выбраны React и Vue [3]. React.js является доминирующей технологией JavaScript и занимает преобладающую долю рынка разработки, поддерживается Facebook, Instagram и сообществом разработчиков. Vue также занимает одну из лидирующих позиций на рынке frontend-разработки, при этом являясь самым молодым веб-фреймворком [1]. Указанная технология позиционирует себя как современный прогрессивный фреймворк и составляет достойную конкуренцию технологиям, разработанным ранее.

В связи с вышеизложенным, возникла потребность разобраться в особенностях выбора технологии при разработке клиентской части приложений, плюсах и минусах использования каждой из них. В качестве критериев для сравнения были выбраны: DOMContentLoaded Event, First Paint, First Contentful Paint, Largest Contentful Paint, Run Microtasks, Parse HTML [2]. Также было проведено исследование по summary-метрикам: loading, scripting, rendering, painting, total (blocking time) и скорость выполнения запросов к API.

Для сравнительного анализа использовались следующие инструменты разработчика: Google Chrome (110.0.5481.177), Chrome DevTools (Performance, Lighthouse). Операционные системы, использованные для проведения эксперимента: macOS Monterey 12.1, Windows 10.

Выводы. В результате эксперимента выявлено, что в приложениях небольшого объема Vue JS работает эффективно. Результаты, полученные в ходе эксперимента, приведены далее.

Сравнительный анализ скорости работы приложений:

Результаты на Vue JS:

First Contentful Paint - 3.6 с

Largest Contentful Paint - 6.7 с

Run Microtasks - 39.125 мс

Parse HTML - 71.61 мс
Loading - 38.5 мс
Scripting – 144 мс
Rendering – 33 мс
Painting – 2 мс
Total – 5201 мс
Скорость выполнения запросов к API - 39.36 мс

Результаты на React JS:
First Contentful Paint - 0.9 с
Largest Contentful Paint - 5.8 с
Run Microtasks - 93.69 мс
Parse HTML - 106.755 мс
Loading - 13.5 мс
Scripting – 560.5 мс
Rendering – 151.5 мс
Painting – 32 мс
Total – 5459.5 мс
Скорость выполнения запросов к API - 41.45 мс

Относительно общей производительности при рендеринге страницы Vue работает быстрее и превосходит React по общему времени отрисовки страницы (total), по скорости выполнения запросов к API, Run Microtasks, Parse HTML, Scripting, Rendering, Painting.

Список использованных источников

1. Волков В.Н., Новиков С.В., Герасимова Н.М., Полонская Т.С. Перспективы использования фреймворка Vue.js для создания гибких пользовательских интерфейсов // Современное программирование. Материалы IV Международной научно-практической конференции. Под общей редакцией Т.Б. Казиахмедова. Нижневартовск, 2022. – С. 69–74.
2. Verbaj, Ylber, "KRAHASIMI I JAVASCRIPT LIBRARIVE – REACT JS ME VUE JS" (2020). Theses and Dissertations. 2038. – DOI 10.33107/ubt-etd.2020.2547. – Текст: электронный.
3. Акулов А.А., Бариев Р.Р. Исследование популярных JavaScript фреймворков для разработки клиентской части веб-приложения // Цифровизация экономики: направления, методы, инструменты. Сборник материалов I Всероссийской студенческой научно-практической конференции. – 2019. – С. 358–362.