

УДК 004.051

ИССЛЕДОВАНИЕ ПРОИЗВОДИТЕЛЬНОСТИ MVC И MVVM- ФРЕЙМВОРКОВ ДЛЯ ФРОНТЕНДА

Воронкова И.О. (Университет ИТМО)
Научный руководитель – Государев И. Б.
(Университет ИТМО)

Введение

В настоящее время происходит активное развитие веб-технологий, в разных отраслях появляется большое количество сайтов, которые пользователи используют в развлекательных, научных и других целях. Сайты компаний привлекают клиентов и служат своего рода рекламой услуг компании, поэтому очень важно, чтобы разрабатываемое веб-приложение было быстрым, стабильным и отзывчивым. Современная веб-разработка не обходится без использования фреймворков, предоставляющих модель MVC (Model-ViewController) или вариации на её основе, например MVVM (Model-ViewViewModel). Они значительно ускоряют процесс разработки и предоставляют разработчикам удобные абстракции для построения интерактивных пользовательских интерфейсов.

В данной работе будет проведен сравнительный анализ производительности MVC и MVVM-фреймворков: Angular и Vue, чтобы определить, какой из них лучше всего подойдет для планируемого проекта.

Основная часть

В статье приведено сравнение различных MVC и MVVM -фреймворков и выделены два самых популярных. По данным GitHub, hh.ru, Stack Overflow Developer Survey и Google Trends для сравнительного анализа были выбраны Angular и Vue.

Проведение эксперимента

Критерии фреймворков:

1. Отправка запросов на сервер
2. Работа с таблицей:
 - a. Добавление 1000 строк
 - b. Удаление всех строк
 - c. Удаление одной строки
 - d. Поменять строки местами
 - e. Обновление всех строк
 - f. Обновить каждую 5-ю строку
3. Печать 5000 элементов

Сравнительный анализ производительности

Размер бандла: Vue имеет меньший размер бандла (minified + gzipped), что означает меньшую нагрузку на клиентскую сторону и более быструю загрузку страниц. (20kb против 100kb у Angular)

Скорость инициализации: Vue.js имеет меньшее время рендеринга (95ms), что делает его подходящим для простых, небольших проектов. Angular же имеет большее время рендеринга из-за сложности своей архитектуры(170ms).

Рендеринг на стороне сервера (SSR): Vue.js имеет встроенную поддержку SSR, что может улучшить производительность при отрисовке на стороне сервера. Angular также поддерживает SSR, но его настройка может быть сложнее.

Обновление компонентов: Vue.js предлагает эффективный механизм для обновления компонентов, что уменьшает необходимость перерисовки всего дерева компонентов. Angular использует мощный механизм обновления, но в некоторых случаях может потребоваться более сложная логика обновления.

Таблица: В последней версии Angular были достигнуты практически идентичные показатели по объёму занимаемой памяти по сравнению со Vue. Так же Angular превзошел Vue по скорости выполнения некоторых операций.

Выводы

В статье были рассмотрены распространенные среди разработчиков MVC и MVVM-фреймворки и выявлены самые используемые из них. Далее проведены эксперименты для выявления самого производительного среди рассмотренных фреймворков.

Angular и Vue – это мощные инструменты, каждый со своими преимуществами и особенностями.

Angular может быть лучшим выбором для крупных, сложных приложений с мощной серверной логикой и строгими требованиями к архитектуре, так как данный фреймворк имеет мощную экосистему: большая команда разработчиков, поддержка компанией Google, обширный набор инструментов и функций. Так же Angular использует мощную систему зависимостей и встроенное управление состоянием. Это удобно для крупных проектов, где требуется строгая структуризация кода и эффективное управление состоянием приложения.

Vue же обычно предпочтительнее для проектов, где важна высокая скорость разработки, простота поддержки и отличная начальная загрузка и производительность отрисовки. Vue имеет более компактные бандлы, что приводит к более быстрой загрузке страницы, а также данный фреймворк известен своей быстрой реакцией на изменения данных и обновление компонентов. Vue предоставляет более легковесный подход к управлению состоянием, что может быть преимуществом для менее сложных проектов.

Но выбор фреймворка зависит не только от производительности, но и от других факторов, таких как сложность проекта, экосистема, сообщество, интеграция с другими технологиями и предпочтения команды разработчиков.

Список литературы

1. Mishra S. P., Srivastava S. K. Web development frameworks and its performance analysis—a review //Smart Computing. – 2021. – С. 337-343.
2. Родионов В. В., Бурханов К. С. СПРАВНЕНИЕ ПРОИЗВОДИТЕЛЬНОСТИ VUE, REACT, ANGULARJS //Актуальные вопросы науки и практики. – 2020. – С. 93–96.
3. Максимов Я. А., Обзор современных программных решений в области измерения производительности клиентской части веб-приложений / Я. А. Максимов, А. И. Мартышкин // Современные наукоемкие технологии. – 2021. – № 12–2. – С. 348–354. – DOI 10.17513/snt.39001. – EDN KAASTH.
4. GitHub [Электронный ресурс] URL: <https://github.com/>.
5. Stack Overflow Developer Survey [Электронный ресурс] URL: <https://survey.stackoverflow.co/2023/#technology>.

Воронкова И.О. (автор)

Государев И.Б. (научный руководитель)