

УДК 004

**АНАЛИЗ СОВРЕМЕННЫХ БИБЛИОТЕК JAVASCRIPT ДЛЯ
РАЗРАБОТКИ ПРИКЛАДНЫХ РЕШЕНИЙ**

Гапон М.В., студент гр. ИТб-182, IV курс

Алешков М.А., студент гр. ИТб-182, IV курс

Научный руководитель: Сыркин И.С., к.т.н. доцент кафедры ИиАПС
Кузбасский государственный технический университет имени Т. Ф.Горбачева
г. Кемерово

В современных реалиях при разработке прикладных решений используются язык программирования JavaScript. Опросы показывают, что 70% среди опрошенных предпочитают его.[1] Такие высокие результаты в опросе неудивительны, так как данный язык программирования прост в освоении, имеет множество библиотек, фреймворков, инструментария, а также различных приспособлений, которые с лёгкостью поддерживаются в JavaScript. При разработке веб-приложения важно учитывать его производительность, для этого необходимо понимать технические характеристики JS-библиотек.

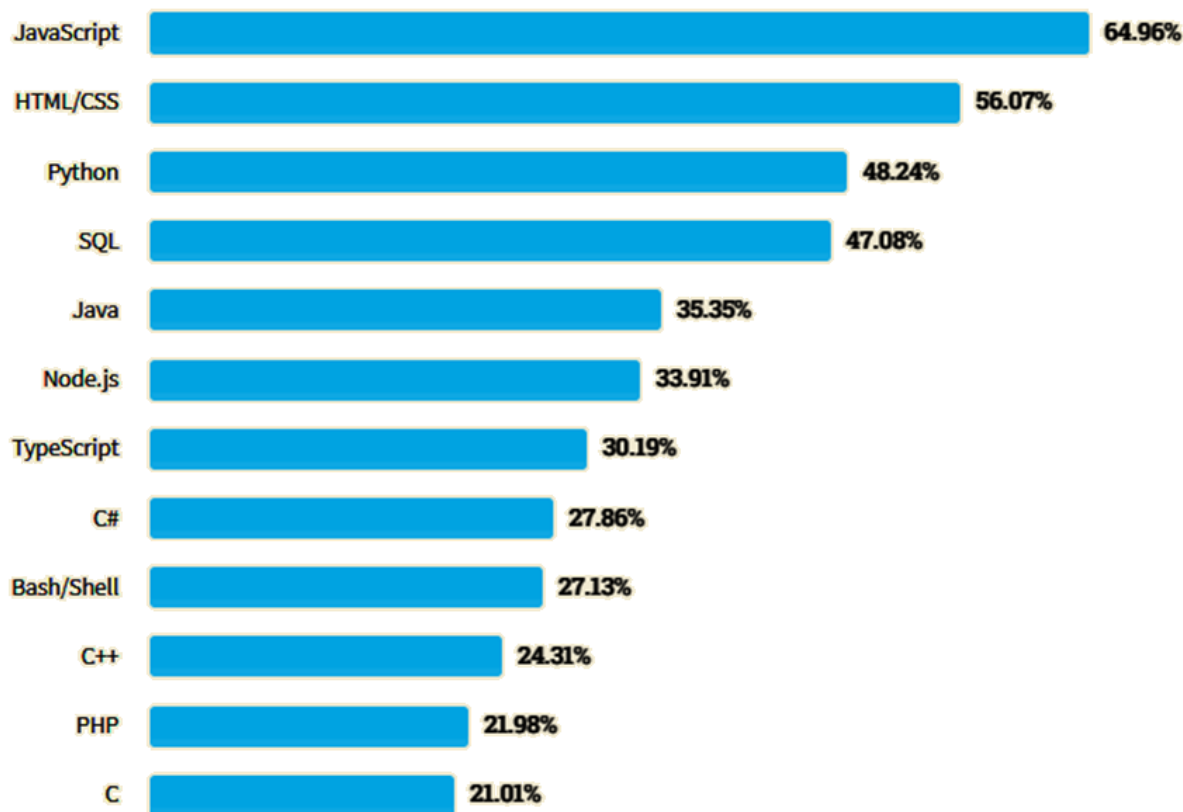


Рисунок 1- Опрос «Какой язык программирования вы используете?»

Для проведения анализа библиотек необходимо простым языком объяснить, что это такое и как в основном это используют. Библиотеки представляют набор многократно используемых функций, объектов и классов. С помощью библиотек происходит ускорение разработки решений с возможностью добавления различного функционала в него. Применение библиотек стало очень доступным, так как для JavaScript существуют менеджеры пакетов с помощью, которых происходит скачивание и развертывание в разрабатываемое решение.

Рассмотрим три самых популярных на основе опросов, библиотеки. [1]
А именно преимущества и недостатки:

1. React.JS
2. AngularJS
3. Vue.JS

React — это декларативная, эффективная и гибкая JavaScript-библиотека для создания пользовательских интерфейсов.[2]

Преимущества:

- С помощью библиотеки можно составлять интерфейсы из более простых элементов.
- React избавляет разработчика от прямого взаимодействия с DOM.
- Библиотека обладает активной поддержкой команды разработчиков, а также обширным сообществом пользователей
 - React обладает открытым исходным кодом.

Недостатки:

- Проекты основанные на React увеличивают размер решений, загружаемых пользователями (~40 kB для пакетов React и React-dom).
- Загруженный код выполняется в браузере, что замедляет запуск решения
- При разработке на React, помимо изучения библиотеки, необходимо привыкнуть к используемой парадигме, что влияет на порог входа.[3]

Angular представляет собой набор библиотек от компании Google для создания клиентских приложений. Прежде всего он нацелен на разработку SPA-решений, то есть одностраничных приложений.[4]

Преимущества:

- Упрощенная двусторонняя привязка данных. Взаимодействия с DOM вложены в простые элементы, что позволяет быстро и просто использовать их в HTML шаблонах.
- Библиотека спроектирована универсально, поэтому с ее помощью можно создать веб-приложение почти любого типа.

- В приоритете минимальное количество кода, поэтому приложения Angular, как правило, компактные и легкие в редактировании

Недостатки:

- Библиотека обладает своей строгой структурой, но несмотря на это спроектирована максимально дружелюбно для пользователей, что ограничивает неопытных пользователей от неоправданно сложных решений.
- Angular не предназначен для манипуляций с DOM при большом количестве данных, по причине использования «грязных проверок» для управления изменениями DOM, в связи с чем при любом изменении переменных обновляется и сам DOM.
- Библиотека плохо взаимодействует с высоконагруженными коллекциями мультимедиа файлов, что влияет на производительность. Данная проблема решается при помощи форм с высоким уровнем пользовательского взаимодействия. [5]

Vue — это библиотека для разработки как простых, так и сложных пользовательских интерфейсов спроектированная по стандартам HTML, CSS и JavaScript

Преимущества:

- Основная идея разработки Vue.js состоит в получении идеального результата при минимальных затратах, так чтобы пользователь мог бы кодировать, используя малое количество строк.
- Vue.js очень просто интегрируется в проекты. Части, собранные с помощью Vue.JS интегрируются в проект, не оказывая отрицательного влияния на всю систему.
- Vue.js — отличный инструмент для разработчиков, потому что имеется достаточно легкодоступный инструментарий.
- Дизайн Vue.js предлагает меньше ограничений и большую гибкость при разработке проекта. [6]

Недостатки:

- Недостаток ресурсов. Vue.js по-прежнему занимает достаточно малую долю рынка по сравнению с React или Angular. По этой причине обмен информацией и документации слабо развит и только начинает развиваться.
- Как было сказано выше, Vue.js очень гибкий и простой в использовании. Это может привести к тому что система будет иметь плохо спроектированную структуру и запутанный для понимания код, поскольку каждый в корпоративной команде может иметь разные мнения о том, как что-то делать.

Рассмотрим технические показатели рассматриваемых библиотек и сделаем анализ на основе этих показателей. [7]

Первым стоит рассмотреть размер JS- кода, который будет передаваться сайтами в сети.

Таблица 1 - Объем JavaScript-кода (Кб), переданного на настольные устройства

Все сайты	1267,3
Vue-сайты	1643,1
Angular-сайты	3283,1
React-сайты	2197,8

Сайты, написанные на Angular, отправляют больше данных на настольные устройства на 159% чем на все сайты. Сайты на React отправляют на 73% больше, чем все сайты. Сайты же на Vue отправляют на 36 % процентов больше

После того JavaScript-кода был передан на устройство, его необходимо запустить. Запуск происходит в главном потоке браузера. Поток отвечает за обработку действий пользователей, построение с последующим выводом макета с применением стилей. Если загрузить главный поток, то это может привести к замедлению загрузки страницы

Таблица 2 - Процессорное время (в миллисекундах) на обработку скриптов, когда используется только одна библиотека

Сайты, построенные только на Vue	9843.8
Сайты, построенные только на Angular	11607.7
Сайты, построенные только на React	24956.3

Angular-проекты, по сравнению со всеми сайтами, тратят на десктопных устройствах в главном потоке на 29% больше времени. В проектах с React аналогичные показатели выглядят как 130%

Дальше проанализируем результаты теста JS-библиотек для определения характеристик производительности, проведенного в браузере Google Chrome. [8]

Name	vanillajs-keyed	inferno-v3.10.1-keyed	vue-v2.5.3-keyed	preact-v8.2.6-keyed	svelte-v1.41.2-keyed	angular-v5.0.0-keyed	react-v16.1.0-keyed	angular-v1.6.3-keyed	ember-v2.16.2-keyed
create rows Duration for creating 1000 rows after the page loaded.	137.8 ± 9.9 (1.0)	150.6 ± 6.8 (1.1)	169.2 ± 3.6 (1.2)	182.8 ± 4.4 (1.3)	182.2 ± 4.7 (1.3)	185.7 ± 7.8 (1.3)	201.2 ± 12.1 (1.5)	222.9 ± 8.1 (1.6)	361.2 ± 23.6 (2.6)
replace all rows Duration for updating all 1000 rows of the table (with 5 warmup iterations).	155.7 ± 5.4 (1.0)	152.3 ± 5.5 (1.0)	161.8 ± 3.9 (1.1)	187.5 ± 7.2 (1.2)	192.9 ± 5.0 (1.3)	179.3 ± 6.5 (1.2)	169.0 ± 4.3 (1.1)	232.3 ± 8.7 (1.5)	238.8 ± 7.7 (1.6)
partial update Time to update the text of every 10th row (with 5 warmup iterations) for a table with 10k rows.	76.5 ± 4.8 (1.1)	92.6 ± 5.0 (1.3)	168.1 ± 7.4 (2.3)	101.1 ± 2.6 (1.4)	72.0 ± 4.6 (1.0)	73.5 ± 4.9 (1.0)	90.9 ± 3.3 (1.3)	87.1 ± 5.3 (1.2)	128.5 ± 3.2 (1.8)
select row Duration to highlight a row in response to a click on the row. (with 5 warmup iterations).	10.8 ± 3.5 (1.0)	13.3 ± 3.1 (1.0)	9.8 ± 2.5 (1.0)	12.5 ± 4.6 (1.0)	9.6 ± 5.9 (1.0)	7.6 ± 4.0 (1.0)	12.4 ± 4.1 (1.0)	10.0 ± 4.7 (1.0)	8.6 ± 3.5 (1.0)
swap rows Time to swap 2 rows on a 1K table. (with 5 warmup iterations).	18.3 ± 4.6 (1.0)	18.8 ± 4.6 (1.0)	19.0 ± 2.8 (1.0)	19.0 ± 2.8 (1.0)	127.1 ± 3.1 (7.0)	118.5 ± 2.8 (6.5)	121.8 ± 4.2 (6.7)	125.9 ± 5.3 (6.9)	128.6 ± 5.0 (7.0)
remove row Duration to remove a row. (with 5 warmup iterations).	43.1 ± 1.6 (1.0)	45.0 ± 2.5 (1.0)	52.5 ± 1.8 (1.2)	46.9 ± 1.6 (1.1)	47.9 ± 1.7 (1.1)	46.1 ± 2.6 (1.1)	51.5 ± 2.0 (1.2)	48.6 ± 2.5 (1.1)	53.6 ± 2.5 (1.2)
create many rows Duration to create 10,000 rows	1,374.5 ± 33.3 (1.0)	1,429.0 ± 32.5 (1.0)	1,521.4 ± 55.7 (1.1)	2,531.0 ± 36.3 (1.8)	1,948.0 ± 51.2 (1.4)	1,682.0 ± 53.1 (1.2)	2,033.7 ± 32.0 (1.5)	2,112.0 ± 77.7 (1.5)	2,406.0 ± 44.6 (1.8)
append rows to large table Duration for adding 1000 rows on a table of 10,000 rows.	217.4 ± 7.3 (1.0)	221.5 ± 7.7 (1.0)	338.4 ± 10.3 (1.6)	384.2 ± 11.5 (1.8)	265.6 ± 11.6 (1.2)	257.6 ± 11.1 (1.2)	271.8 ± 9.9 (1.3)	371.6 ± 60.4 (1.7)	432.3 ± 31.4 (2.0)
clear rows Duration to clear the table filled with 10,000 rows.	177.1 ± 10.2 (1.0)	328.9 ± 8.6 (1.9)	240.9 ± 11.4 (1.4)	238.2 ± 3.5 (1.3)	237.7 ± 4.2 (1.3)	360.3 ± 16.4 (2.0)	224.4 ± 6.0 (1.3)	517.8 ± 62.0 (2.9)	262.3 ± 10.2 (1.5)
slowdown geometric mean	1.01	1.13	1.28	1.31	1.46	1.48	1.50	1.78	1.91

Рисунок 2 - Результат теста производительности

По результатам теста делаем следующие выводы:

1. При создании строк (строки Create rows & Create Many rows) сайты разработанные на Vue JS быстрее создают строки после загрузки страницы, чем сайты, построенные на основе других рассматриваемых библиотек, но разница во времени выполнения незначительна)
2. В случаях, когда происходит обновление строк (строка partial update) AngularJS выполняет запрос обновления быстрее, в то время как Vue наоборот имеет проблемы, которые замедляют процесс
3. Выбор строки и удаление(select/remove) у всех библиотек происходит одинаково быстро

4. При выполнении операциях перемещений строк (swap rows) Angular и React сайтам нужно выделить больше времени на выполнении, чем Vue-сайту

По результатам рассмотрения библиотек и проведенного технического анализа, для реализации наших проектов была выбрана JavaScript библиотека React по следующим причинам:

- Библиотека очень популярна и имеет готовую документацию, составленную опытными разработчиками.
- Обладает обширным сообществом пользователей, у которого всегда можно будет подчерпнуть необходимые знания
- Простота в интеграции в проект новых разработчиков
- Легкость составления компонентов информационной системы
- Удовлетворимый размер кода для загрузки ИС
- Удовлетворяющим требованиям заказчика по скорости взаимодействия пользователя с системой

В заключении необходимо отметить:

1. Библиотеки должны давать разработчикам качественную базу для создания веб-проектов и обладать встроенными ограничениями, которые усложняют создание чего-то такого, что эти ограничения нарушают.
2. При выборе библиотек для разработки надо учитывать, что нет иногда необходимости в многофункциональных библиотеках, так как их возможности не всегда будут нужны при разработке
3. Не всегда будет лучшим решением использовать новые, еще разрабатываемы библиотеки в ходе разработки проектов.

Список литературы

1. Система вопросов и ответов о программировании [Электронный ресурс]. -Режим доступа: <https://insights.stackoverflow.com/survey/2021#most-popular-technologies-database>, свободный. (Дата обращения: 20.03.2022)
2. Документация ReactJS [Электронный ресурс]. -Режим доступа: <https://ru.reactjs.org/tutorial/tutorial.html>, свободный. (Дата обращения: 20.03.2022)
3. Бесплатный образовательный проект о информационных технологиях [Электронный ресурс]. -Режим доступа: <https://academy.yandex.ru/posts/что-такое-react-i-kak-ego-osvoit>, свободный. (Дата обращения: 20.03.2022)

4. Сайт о программировании [Электронный ресурс]. -Режим доступа: <https://metanit.com/web/angular2/1.1.php>, свободный. (Дата обращения: 20.03.2022)
5. Проект посвященный web-разработке [Электронный ресурс]. -Режим доступа: <https://webformyself.com/18-sovetov-po-povysheniyu-proizvoditelnosti-vvedenie-v-angular-js/>, свободный. (Дата обращения: 20.03.2022)
6. Ресурс о компьютерных технологиях [Электронный ресурс]. -Режим доступа: <http://fl-it.ru/kakov-plyus-i-minus-ispolyzovaniya-vue-js.html>, свободный. (Дата обращения: 20.03.2022)
7. Блог с элементами новостного сайта о информационных технологиях, бизнесе и интернете [Электронный ресурс]. -Режим доступа: <https://habr.com/ru/company/ruvds/blog/499280/>, свободный. (Дата обращения: 20.03.2022)
8. Блог затрагивающий разные сферы жизнедеятельности [Электронный ресурс]. -Режим доступа: <https://medium.com/nuances-of-programming/%D1%81%D1%80%D0%B0%D0%B2%D0%BD%D0%B5%D0%BD%D0%B8%D0%B5-%D0%BF%D1%80%D0%BE%D0%B8%D0%B7%D0%B2%D0%BE%D0%B4%D0%B8%D1%82%D0%B5%D0%BB%D1%8C%D0%BD%D0%BE%D1%81%D1%82%D0%B8-js-%D1%84%D1%80%D0%B5%D0%B9%D0%BC%D0%B2%D0%BE%D1%80%D0%BA%D0%BE%D0%B2-%D0%BD%D0%B0-realworld-demo-1c14000ec1c6>, свободный. (Дата обращения: 20.03.2022)