International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 04 | Apr 2020 www.irjet.net p-ISSN: 2395-0072

Analysis of Front-end Frameworks for Web Applications

Vinuta Hutagikar[1] and Vinay Hegde[2]

[1]Department of Computer Science and Engineering, R.V College of Engineering, Bengaluru, Karnataka, India
[2]Professor, Department of Computer Science and Engineering, R.V College of Engineering, Bengaluru, Karnataka, India

---

Abstract - *The most fundamental move and standpoint of a software development is to select the right front end framework. The market has a wide variety due to the wide range of problems that developers face every day. And the number of new front-end framework increases considerably. A web application can be built according to design with various frameworks available taking into account all the constraints and feasibility. This paper explains the benefits and drawbacks in relation to the basic aspects and unique characteristics of frameworks. Also, provides a systematic review of the literature on front end frameworks for single page applications (SPA) . Using a systematic literature review as methodology, this paper presents an overview of the front end frameworks identified in the literature, the key features comprising these frameworks. The requisite characteristics were analyzed in the three most common frameworks: Vue.js, Angular and React.*

Key Words: Front end framework, web application, SPA, feasibility.

1. INTRODUCTION

There has been a great demand in recent years for extremely advanced and dynamic web applications, to replace the old desktop application in all areas. The development of web applications is often based on the use of a certain framework and sometimes combination of frameworks for better performance.

Nowadays, single-page applications have become quite common, despite their vast capacity for creating web and mobile apps.

The aim of the paper is to demonstrate the suitability of the frameworks available for the development of SPA web applications. Also, to identify main features that enable more personalized development of SPA web applications.

2. RELATED WORK

An efficient framework is essential for developing single-page applications. In addition, the frameworks used should be able to cope with the dynamic architecture involved in the creation of single web applications for better user experience and interactivity. SPA is a modern approach to the creation of web applications which is mostly used by applications with less content [1].

A Single Page Application, or SPA, is a type of web application where browsing is made feasible without loading entire page frequently since codes like HTML, CSS, JavaScript, etc. are dynamically loaded only once. SPA has only one page for all parts of the user interface which serves as a "shell." All subsequent components are loaded dynamically and independently of the shell, without reloading the entire page, giving the user the illusion that the page has changed. The SPA work period continues and focuses on the user's work in the user interface. SPAs use AJAX when sending a request to a server and receive data as a response (usually in JSON9) and provide parts of HTML to display the provided data. After the data is received from the server, the client side must form the received content and view it at a particular location in the shell[2].

3. ANALYSIS METHODOLOGY

3.1 Analyzed frameworks

Comparison would be made of the three most common JavaScript front end Frameworks, whose prominence continues to rise. The following figure displays the rating of all accessible JavaScript frameworks

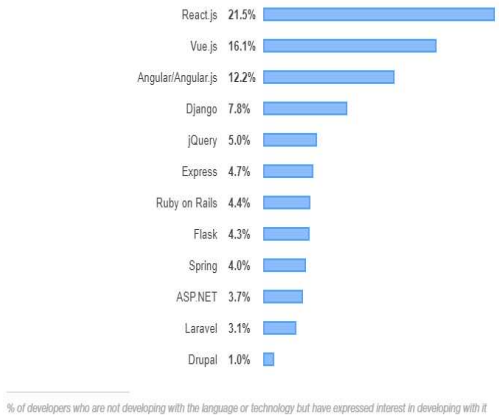

Fig -1

Source: https://existek.com/blog/top-front-end-frameworks-2020/

Analysis (Fig-1) was performed among well-known JavaScript front end frameworks, and it was identified that only the three listed ones are currently gaining prominence while the others are stagnating[3].

- Vue.js
- React
- Angular

The following are several major advantages and drawbacks of these frameworks

3.11 Vue.js

Vue.js is a systematic framework for the development of user interfaces. Unlike a monolithic architecture such as Angular, Vue was designed to be incrementally user-adoptable.

Thanks to its functionality, fairly simple learning curve, and ability to build powerful, quick, and sophisticated single-page applications, it's one of the most starred JS frameworks on Github.

Advantages:

- Vue is fast and extremely light. It offers benefits that include greater efficiency for limited-resource computers, increased usability for users with poor connectivity and a better overall user experience.
- Vue.js helps to build relatively broad modular models that can be rendered without any additional time allotted to it due to its basic architecture.
- Its reactivity is one of the strongest features of Vue.js . Linking data between HTML and JS code has never been easier.
- Vue is treating two-way dynamic data-binding like a charm. When the data is updated, the DOM should always be modified accordingly which is perfect for SPA.

Disadvantages:

- Compared to Respond or Angular, Vue.js only has a relatively low market share, which means that information exchange in this framework is only in the early stages.
- Risk of being over flexible while integrating with large projects due to lack of resources.

3.12 React

React is a JavaScript library used to build stateful and reusable UI components. open sourced by Facebook in 2013, and is perfect for setting up modern single-page applications of any size and scope[4].

Advantages:

- React uses a virtual DOM that achieves optimum efficiency by re-rendering nodes as desired.
- Server side rendering is yet another important advantage of this framework for some type of implementations such as content focused applications.
- Minimizes user's resource load by offering support for bundling and tree shaking.
- Functional programming of react helps in creating reusable code.
- Relatively better SEO in comparison with angular and vue.js .

Disadvantages:

- Need to import libraries for state and model as React doesn't implement MVC.
- React moves away from class-based modules and can be an obstacle for Object Oriented Programming (OOP) for developers to feel more relaxed.

3.13 Angular

This framework was first published in 2009, and was renamed AngularJS. Nonetheless, currently this term only reflects updates previous to version 2.0 – the later versions are just named Angular. It is built in typescript.

Advantages :

- Due to less bandwidth being used and as entire page not be refreshed as the user navigates, application works faster.
- Angular replicates changes made to the model instantly into views because of two way data binding, in an simple, effective and intuitive manner.
- MVVM (Model-View-ViewModel) which enables developers to operate with the same collection of data separately on the same application.
- Structure and architecture built especially for better scalability of projects
- Dependency injection to the components contributes to the modularity of the application.

Disadvantages:

- While Angular having various structures(Injectables, Components, Pipes, Modules etc.) in comparison with React and Vue.js have only component , the former is harder to learn.

- Constantly updating: new, major improvements are always made. This can create problems for developers when it comes to adapting to them.

3.2 Framework analysis

Following factors were used for analysis of the three frameworks to compare their performance side by side:

- **DOM manipulation**: It lets us decide which architectures are ideally adapted for highly complex applications that need a lot of interaction with the DOM [3].

- **Startup time:** This factors give us an idea of which frameworks are feasible when faster initial load time is needed [3].

- **Data Allocation**: It demonstrates which frameworks function well for data — for example, executing bulk operations such as reading or writing thousands of documents from the database[3].

Colors in the images have a meaning: greener color indicates that the system is doing better than another one. The red / yellow color indicates bad results on the given framework.

Most of the comparisons displayed are keyed models, and all time units are represented in milliseconds.

DOM manipulation

Duration in milliseconds ± 95% confidence interval (Slowdown = Duration / Fastest)

Name Duration for...	angular-v8.0.1-keyed	react-v16.8.6-keyed	vue-v2.6.2-keyed
create rows creating 1,000 rows	123.0 ±3.8 (1.00)	137.4 ±3.7 (1.12)	136.9 ±3.0 (1.11)
replace all rows updating all 1,000 rows (5 warmup runs).	115.6 ±1.4 (1.07)	108.7 ±1.6 (1.00)	108.3 ±1.5 (1.00)
partial update updating every 10th row for 1,000 rows (3 warmup runs). 16x CPU slowdown.	114.0 ±5.2 (1.00)	136.7 ±3.6 (1.20)	200.1 ±6.0 (1.76)
select row highlighting a selected row. (5 warmup runs). 16x CPU slowdown.	26.0 ±2.5 (1.00)	30.4 ±3.5 (1.17)	116.7 ± 17.1 (4.50)
swap rows swap 2 rows for table with 1,000 rows. (5 warmup runs). 4x CPU slowdown.	337.7 ±2.6 (4.89)	344.5 ±4.5 (4.98)	69.1 ±6.2 (1.00)
remove row removing one row. (5 warmup runs).	36.4 ±0.6 (1.00)	39.7 ±0.5 (1.09)	44.8 ±1.7 (1.23)
create many rows creating 10,000 rows	1,211.7 ± 30.2 (1.04)	1,443.8 ± 36.0 (1.24)	1,161.5 ± 38.2 (1.00)
append rows to large table appending 1,000 to a table of 10,000 rows. 2x CPU slowdown	247.5 ±7.8 (1.00)	263.5 ±4.0 (1.06)	255.8 ±3.3 (1.03)
clear rows clearing a table with 1,000 rows. 8x CPU slowdown.	232.5 ± 13.7 (1.90)	122.4 ±4.4 (1.00)	135.3 ±3.7 (1.11)
slowdown geometric mean	1.30	1.31	1.32

Source:https://blog.logrocket.com/angular-vs-react-vs-vue-a- performance-comparison/

Angular appears to do well when it comes to DOM manipulation, save for a few situations in which it performs worse. But, generally speaking, it functions best when it comes to this specific type of operations.

Startup time

Startup metrics (lighthouse with mobile simulation)

Name	angular-v8.0.1-keyed	react-v16.8.6-keyed	vue-v2.6.2-keyed
consistently interactive a pessimistic TTI - when the CPU and network are both definitely very idle. (no more CPU tasks over 50ms)	2,703.7 ± 0.4 (1.24)	2,353.2 ± 0.5 (1.08)	2,177.5 ± 0.3 (1.00)
script bootup time the total ms required to parse/compile/evaluate all the page's scripts	52.0 ± 70.5 (3.25)	16.0 ± 0.0 (1.00)	16.0 ± 0.0 (1.00)
total kilobyte weight network transfer cost (post-compression) of all the resources loaded into the page.	295.5 ± 0.0 (1.40)	260.7 ± 0.0 (1.24)	211.0 ± 0.0 (1.00)
slowdown geometric mean	1.78	1.10	1.00

Source: https://blog.logrocket.com/angular-vs-react-vs-vue-a-performance-comparison/

In this case, Vue.js is the absolute winner. The relatively small size of framework saves a lot when it comes to start-up time.

React has very similar results as Vue.js and angular on the other hand suffers.

Memory allocation

Memory allocation in MBs ± 95% confidence interval

Name	angular-v8.0.1-keyed	react-v16.8.6-keyed	vue-v2.6.2-keyed
ready memory Memory usage after page load.	4.8 ±0.0 (2.22)	2.2 ±0.0 (1.03)	2.2 ±0.1 (1.00)
run memory Memory usage after adding 1000 rows.	9.2 ±0.0 (1.34)	6.9 ±0.0 (1.00)	7.0 ±0.0 (1.03)
update eatch 10th row for 1k rows (5 cycles) Memory usage after clicking update every 10th row 5 times	9.5 ±0.0 (1.28)	8.0 ±0.0 (1.08)	7.4 ±0.0 (1.00)
replace 1k rows (5 cycles) Memory usage after clicking create 1000 rows 5 times	9.8 ±0.1 (1.29)	8.8 ±0.0 (1.15)	7.6 ±0.0 (1.00)
creating/clearing 1k rows (5 cycles) Memory usage after creating and clearing 1000 rows 5 times	6.5 ±0.0 (1.75)	4.6 ±0.0 (1.25)	3.7 ±0.0 (1.00)
slowdown geometric mean	1.54	1.10	1.01

Source: https://blog.logrocket.com/angular-vs-react-vs-vue-a-performance-comparison/

In these groups, Angular is certainly slower than Vue and React. The latter two do very well, confirming the impression that there is no very substantial gap between these two frameworks when it comes to performance. However, Vue is winner in this situation.

4. DISCUSSION

The analysis show that the different frameworks function well based on the following application criteria:

Vue.js is more suitable for applications:

- Light weight applications
- When there is requirement to integrate into existing applications.
- Where speed is critical parameter.

React is feasible framework for applications:

- That require high speed
- Applications that require versatility
- better SEO

Angular is more suitable for:

- Applications with very dynamic content
- Enterprise grade applications and large applications.

5. CONCLUSION

This paper examined the appropriateness of the front end technologies for the creation of SPA Web applications. The front end frameworks Angular, Vue.js and React were described and their advantages and disadvantages. The criteria for selection of specific framework were clearly identified.

According to the results, above mentioned three frameworks are suitable for SPA development depending on application goals and feasibility.

REFERENCES

[1] Dimi (2017) multi page vs single page web applications, https://thmefusse.com/ single page vs multipage design/

[2] Marin Kaluza and Bernad Vukelic "Comparisions of front end frameworks for web applications", vol 6 (2018), pp-261-286

[3] Framework performance comparison (2019) – angular vs react vs vue.js, https://blog.logrocket.com/angular-vs-react-vs-vue-a-performance-comparison/

[4] What to choose(2020) – react,angular or vue.js - https://medium.com/@TechMagic/reactjs-vs-angular5-vs-vue-js-what-to-choose-in-2018-b91e028fa91d

[5] Complete comparison guide (2020) - https://medium.com/front-end-weekly/react-vs-angular-vs-vue-js-a-complete-comparison-guide-d16faa185d61

[6] YongKang Xing, Yong, YongYao Lai "Research and analysis of front end libraries and frameworks", Conference: the 2019 11th International Conference, DOI: 10.1145/3313991.3314021