iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** IV **Month of publication:** April 2022

DOI: https://doi.org/10.22214/ijraset.2022.41254

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue IV Apr 2022- Available at www.ijraset.com

Research and Analysis of the Front-end Frameworks and Libraries in Web Development

Arnav Awasthi[1], Shubham More[2], Warren Viegas[3]
[1, 2, 3] *Xavier Institute Of Engineering, Department of Information Technology, Mumbai University, Mumbai, India*

Abstract: ***With the rapid expansion of online technology in recent years, there is a strong trend that Hypertext Markup Language (HTML)5 will become a global web consortium and will lead the front-end. to take stage in the history of the internet There are, however, a slew of front-end development frameworks to choose from., Angular, and are examples of libraries. How to Choose appropriate framework or library to launch an e-business and expand its react. It becomes a priority procedure to the user experience.in terms of web development This paper begins with an introduction a list of the most popular frameworks and libraries in the field of front-end development and web performance analysis services. This paper examines the research findings from a variety of perspectives. The advantages and disadvantages of each framework will be described in this study., the study concludes with a summary, the contributions, and finishes by speculating on the future of front-end development***
Keywords: Front-end; JavaScript; Web Development; e-Business; HTML5.

I. INTRODUCTION

A. Preface

With the fast advancement of internet technology over the last decade, customers have become increasingly reliant on e-Business to carry out daily tasks such as shopping, obtaining a mortgage, and filing taxes. HTML5 technologies come and modify the entire internet development ecosphere as an innovation, which is one of the most important causes for this result. HTML5 is a markup language that is used to create layout and render information on global websites. In comparison to previous HTML standards, HTML5 adds and improves several semantic components such as footer>, aside>, and nav> to clearly define the web structure, and it aids web developers in building their websites under strict guidelines. HTML5 also provides new components to application programming interfaces (APIs) (APIs).

The <canvas> element, for example, allows the website to access the canvas portion of the mobile phone . Web programmers can create a website with more complex operations using HMTL5's sophisticated access features. Although HTML5 introduces numerous advancements, it still has the drawback that any published HTML version's rendering efficiency is extremely low, even when compared to FLASH. Google created the Chrome V8 engine in 2008, which appropriately tackles the issue of HTML5 bringing JavaScript to the forefront.

Prior to the release of Chrome V8, JavaScript's primary job in a website was to interact with Cascading Style Sheets (CSS) to provid e a better user interface and to handle routine script actions like for m validation. Because Chrome V8's JavaScript engine is more t han 56 times quicker than any version of Internet Explorer (IE) the Chrome V8 look redefines JavaScript.

Traditional web browsers often create JavaScript by parsing byte- code and compiling the complete web project to generate the code, which is then executed from a file system. As a result, its JavaScript execution time is significantly longer than that of compiled languages like Java and C++.

The optimised solution for the V8 engine uses inline caching technology to boost speed without using traditional methods.

Following the release of the V8 engine, JavaScript may be able to operate in a similar manner as good as Java or C++ in terms of performance. As a result, a web project can match the speed of classic desktop apps. Because of the superiority of the V8 JavaScript engine, several JavaScript platforms based on the V8 engine have appeared. This has ushered in a new era in the history of internet development.

In the year 2009, Node.js was released, which works in conjunction with the V8 JavaScript engine. As a development environment Node.js is a programming language that allows developers to enhance their capabilities. JavaScript's vision is that it can't only run a simple script in a browser website, but also in the development of an event-driven server-side application that is easy to use. Despite the fact that Node.js was released, Many new JavaScript frameworks appeared nine years ago. and have an impact on the evolution of the internet. The following section discusses the trace on the major front-end frameworks and libraries.

©IJRASET: All Rights are Reserved

B. Front-end Frameworks and Libraries

Because of the V8 engine's innovation, there are various front-end frameworks and libraries based on JavaScript. We collect usage data from Stack overflow and Github, the world's largest Git-repository hosting service, in order to uncover prominent front – end frameworks and libraries that meet industry standards. The usage statistics on Github and Stack overflow can represent the preferences of worldwide front-end developers for various front-end frameworks and tools.

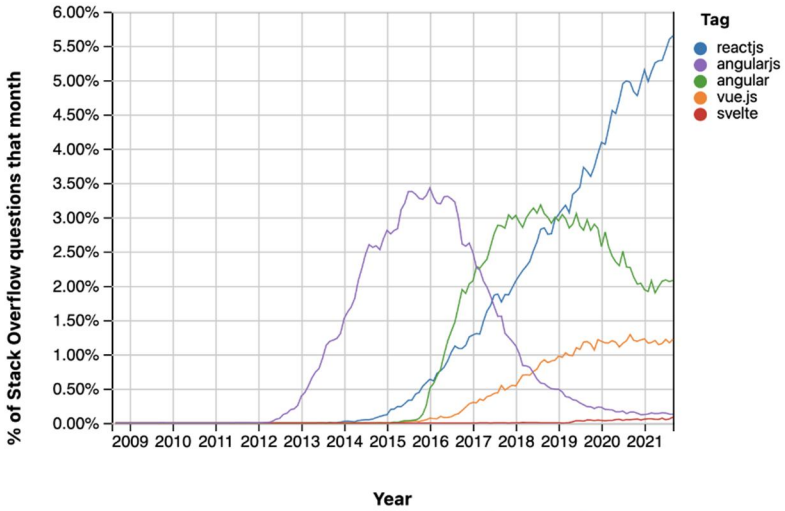


Figure 1 Front-end framework usage data

The graph above shows that React remains in the dominant position. Angular 2 retains the second position whereas Angular 1 and Vue rank relatively lower than the other two in terms of popularity.

1) Vue.js

Evan You created Vue.js, a prominent JavaScript ES6-based open source framework, in 2014. The first goal of Vue's development is to enable responsive data binding and UI components using a simple Application Programming Interface (API). Although Vue is intended for use in single –page applications, it is often regarded as having limited functions and being difficult to adopt in commercial applications. However, the open source community has created a strong third-party supporting library and packages to help Vue drive complex single-page applications with routing, state management, and build tooling.

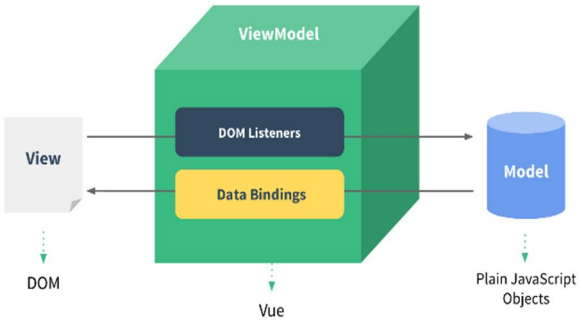


Figure 2 Vue.js Data driven concepts

Vue.js features three components to process data, as shown in Fig. 2. These portions are View, View Model, and Model. The websit e content is displayed in the View area, which is a displayed DOM. DOM listeners and Data Bindings are found in the View Model section. It acts as a data intermediary between the View and the Model. When the data processing in View is triggered by the users Vue will monitor and update the Model section's data using DOM listeners.

Vue will use DOM binding to adjust the website content appearance when Model's data is updated. In brief, Vue achieves two-way binding by using one-way binding with DOM listeners.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

2) Angular 1 & 2

Angular is a well-known open source front- end web application framework based on JavaScript ES5 that was created by Google in 2010 .Angular's primary development goal is to make it easier for web designers to create a persistent web form . As the front end history progresses, Angular's position evolves to match more development criteria, allowing web developers to create more complex apps using the Angular framework. However, because of its basic design model, Angular has fallen significantly behind competing f ront-end frameworks in recent years. In order to bring Angular up to date, Google published its second version, Angular 2, in 2016, which was totally redesigned by the Google development team.

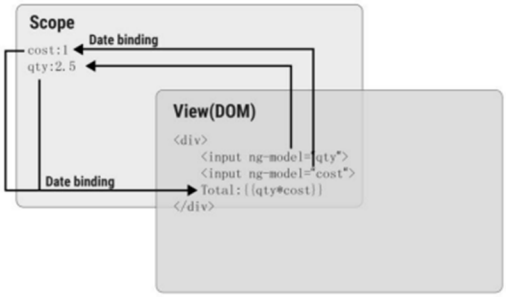


Figure 3 Angular 1 two-way data binding

The main concept of Angular 1 is two-way data binding in web browsers, which significantly reduces the back-end data processing burden on web servers. Figure 2 shows how Angular 1 handles data binding. The custom tag attributes have been incorporated in JSON, and Angular 1 uses them as directives to link input or output elements of the website page to a model represented by Scope. When users take interactive activities on websites, the values of certain JavaScript variables are updated from dynamic JSON resources, and the data is submitted to the server. The two-way data binding of Angular allows whole interactions to be performed in web browsers, avoiding the need for website changes to wait for data processing from the back-end server and directly rendering the updated data in front-end by HTML. Consequently, the HTML rendering speed can be faster without waiting for back-end response due to Angular 1 technology

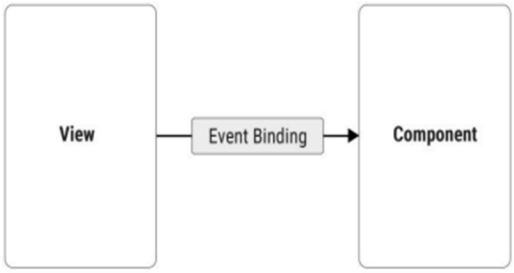


Figure 4 Angular 2 one-way data binding

Angular 2 has completely redesigned its concept and optimized Angular 1's binding process as compared to Angular 1. To begin with, Angular 2 removes the template directive and controller. It combines the two pieces using a new module called 'Component' . Angular 2 provides event binding as a one-way binding from view to component, which was not available in Angular 1.Third, Angular 2 switches from a JavaScript-based language to a TypeScript-based language , which is a strict syntactical superset of JavaScript developed by Microsoft. Fourth, Angular 2 monitors interactive actions with zone.js rather than Scope . Last but not least, Angular 1 was designed for desktop web apps with limited support for mobile platforms, whereas Angular 2 favors mobile devices. By optimizing and integrating different features, Angular 2 has smaller size and faster speed which is essential in mobile development

3) React.js and React Native

In order to establish Facebook and Instagram website with better user experience, Facebook developed the React JavaScript library . Due to React's powerful features, Facebook released React as an open source JavaScript ES6 based library to global developers and companies in 2013 . Besides, Facebook also launches React Native to develop a mobile application with React under major mobile platforms such as IOS and Android in 2015.

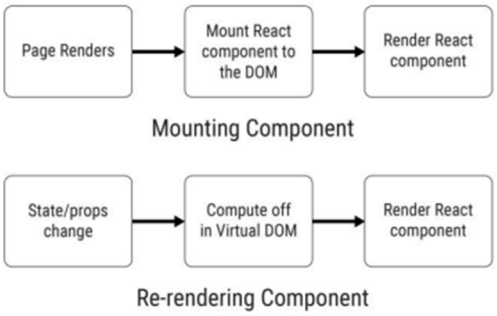


Figure 4 Mounting components in React & Re-rendering components in React.

Figure 5 shows how React technology can be used to render a page. The Document Object Model is where React stores website content as distinct components (DOM). JavaScript will be used to render the component in the browser. The JavaScript rendering speed will be faster than that of traditional dynamic webpages with Chrome V8 technology. Another React's core wisdom is creating a virtual DOM . When a user refreshes the data or visits other subpages on a typical HTML website, the page is completely re-rendered. The re-rendering procedure consumes more browser resources and slows down the webpage. React, in contrast to the old method, takes a different approach to the problem by creating a virtual DOM with one way data binding. React initially produces an updated virtual DOM, then compares the virtual DOM to the visible DOM. It will re-render the actual transform part after it summarises the differences in each component, but other uniform parts will not be re-rendered. The virtual DOM's most major benefit is that the website becomes faster when using the React framework. Another advantage is that React encourages developers to design UIs according to module standards. For example, there are two websites with a text form that perform the same function but have distinct features. Due to the changes in the text forms, when the customer clicks a link and visits another page, the page will still re render the text form. If the development team creates a UI module standard and then picks the text form from the module to apply to various pages, the text form will not be re rendered since React will identify that the two forms are identical and will not alter them. In short, React not only revolutionises front end programming, but it also establishes a new standard in user interface design and development.

II. FRONT-END SOLUTION ANALYSIS

React, Angular, and Vue are unquestionably prominent frontend development alternatives in modern web development. They do, however, have distinct qualities and concepts. It means that different commercial criteria must choose an appropriate framework or library to get the most out of their applications.

A. Data Processing

Data processing is an important aspect of front-end development since the efficiency and quality of data processing determine the user experience when accessing the application.

	Angular 1	Angular 2	React	Vue
Data Binding	Two-way	Two-way and one-way	One-way	One-way /Two-way

Table 1 Data Binding in front-end frameworks

Table 1 shows how each framework and library handles data binding. Vue may achieve two-way binding via DOM listeners or one-way binding without them. The choice of one-way or two-way data binding is the most major difference between React and Angular 1. React, as comparison to Angular 1, has a more sophisticated data flow to check the difference between the virtual DOM and the shown DOM. However, using a two-way method between a component and a view may cause the component to transition into unexpected states as a result of conflicting data from multiple sources, whereas using a one-way method can avoid conflict issues in multiple data sources, particularly in event-based situations. As a result, the Angular 2 team has improved the concept of developers being able to leverage both one-way and two-way binding.

B. Volume and Performance

A larger volume indicates that the framework or library has more features and functions, but it will take longer to load a framework or a library.

Activity	Angular 2	React	Vue
Ready-memory	4.8 ± 0.0 (1.4)	4.5 ± 0.1 (1.3)	3.8 ± 0.0 (1.1)
Run-memory	10.9 ± 0.1 (2.7)	9.7 ± 0.1 (2.6)	7.5 ± 0.1 (1.9)

Table 2 Front-end frameworks and library memory allocation performance

Angular 2 has the largest volume, with 143 kilobytes (KB), Vue with 23 KB and React with 43 KB. Because of its massive volume, Angular 2 includes more advanced and extensive functions and capabilities. However, due to its complex structure, Angular 2's running speed may be inferior to that of React or Vue, particularly when it comes to memory allocation. Table 3 demonstrates that Angular 2 takes longer to prepare and run the memory, whereas Vue takes less time because of its flexibility and efficiency.

C. Language-based

Language-based considerations are also significant because different languages have varied situations in terms of project development, such as a learning curve and efficiency.

	Angular 1	Angular 2	React	Vue
Language-based	JavaScript ES5	TypeScript	JavaScript ES6	JavaScript ES6

Table 3 Language-based in front-end frameworks and library

Table 3 depicts the present state of language-based frameworks and libraries. React and Vue are built on JavaScript ES6, which is the most recent JavaScript industry standard from 2015. The previous version of JavaScript, ES5, is used by Angular 1. TypeScript is used by Angular 2 to deliver a better type inference experience and to decrease all type problems in online apps. In addition, TypeScript helps developers get rid of the old JavaScript programming format by streamlining the language structure. TypeScript, on the other hand, has a small user base. There's a chance TypeScript will become obsolete if a new syntactical tight superset of JavaScript emerges. Because it is the industry criterion in producing JavaScript, JavaScript ES6 can be modified but not removed.

D. Technical Support

Technical assistance is also required since greater support may help build stronger relationships with developer communities, which is critical for the framework's reputation to grow. React has excellent technical support and a very reliable API.

The official scripts to aid developers in doing the related update make upgrading and immigration relatively straightforward. In a nutshell, React provides long-term technical support. Angular has similar functions, but its API isn't as robust as React's.

APIs from previous versions have been removed. Despite the fact that Vue includes an easy migration and update system across versions, the official team has no plans to improve it owing to funding constraints.

E. Online Business Solutions

Based on the preceding examination of many features of front-end frameworks and libraries, it is reasonable to conclude that each framework or library has its own set of strengths and weaknesses. In terms of data processing, Angular 2 offers the best one-way and two-way binding solution. Furthermore, the Google development team's official technical support is consistent and dependable. However, due to its numerous functionalities, its volume is too large to play an adequate running performance, and its based language has a small community. As a result, Angular 2 is well suited to large-scale e-Business solutions that demand complex functionality and advanced data processing methods.

React's exceptional efficiency in rendering updated DOM, as well as its rich technical support and long-lasting API, allow developers to eliminate updating and immigration concerns. Furthermore, after learning React, developers can construct React Native mobile applications immediately. However, because React is a JavaScript library with a smaller volume than Angular 2, it cannot provide comprehensive functionalities and developers must deploy on their own. Typical requests for re-development of more customizable features and functionalities with quick rendering speed come from social media and communication applications. As a result, they are React's potential Clients.

In data processing, Vue offers both two-way and one-way binding options. Its minimal volume happens in the most efficient rendering and processing as compared to Angular 2 and React. Although Vue has a major flexibility advantage in front-end development, its technical support is unreliable due to its development team's scale limitations, as well as unexpectedly changed official plans. Furthermore, because of its modest size, it only has the most basic functionality. Vue is appropriate for small and medium web projects that require flexibility and simplicity in development, as well as the fastest data processing speed.

III. CONCLUSION

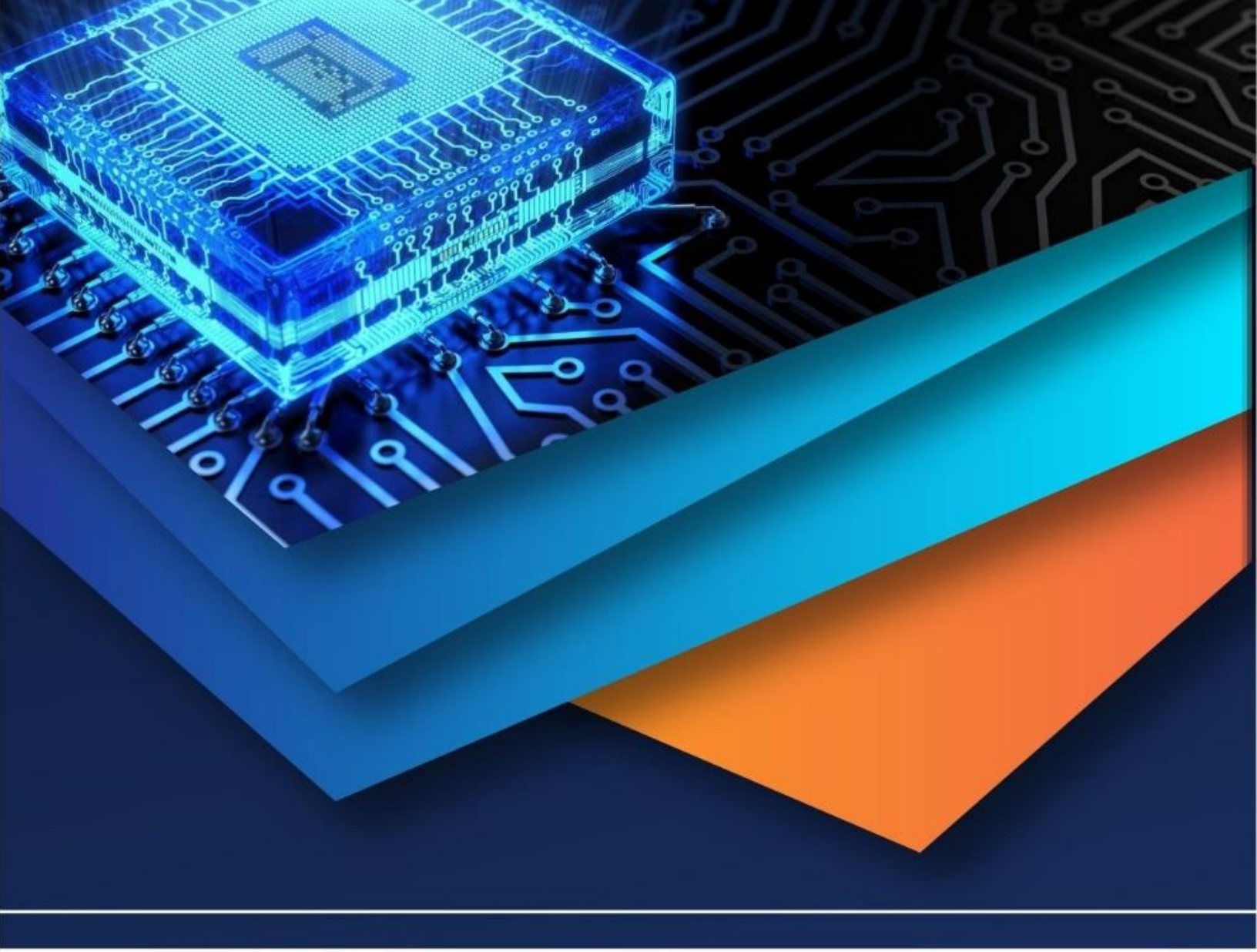
This paper discusses three possible front-end development frameworks and libraries for constructing online apps, as well as prospective web application development solutions. By comparing React, Angular 2, and Vue in many aspects such as data binding, language-based, technical support, volume, and performance, we may draw some conclusions. It is feasible to conclude that Angular 2 has the most extensive functions and features, making it ideal for large commercial projects, particularly in the eBusiness sector. Live streaming, communication, blogging, and small to medium size applications are all good candidates for React and Vue.

A UI framework must also be used when developing a whole front-end component to demonstrate expert UI design. Our research field will be expanded in the future to include more front-end development methodologies and an analysis of their functioning principles to develop web applications.

REFERENCES

[1] Stefankrause Developer Team., 2018, December. Results for js web frameworks benchmark. From https://www.stefankrause.net/js-frameworksbenchmark6/webdriver-ts-results/table.html
[2] M. R. Solanki, A. Dongaonkar, A Journey of human comfort: web1.0 to web 4.0, International Journal of Research and Scientific Innovation (IJRSI), Volume III, Issue IX, pp. 75-78, 2016
[3] Angular Official Blog., 2018, July. AngularJS to Angular Concepts: Quick Reference From https://angular.io/guide/ajs-quick-reference
[4] Vue Official Blog., 2018, July. Reactive Data Binding Research. In Science and Information Conference From https://v1.vuejs.org/guide/overview.html
[5] MDN Web Docs. A re-introduction to JavaScript. Retrieved from https://developer.mozilla.org/en-US/docs/Web/JavaScript/A_reintroduction_to_JavaScript.
[6] React. Virtual DOM and Internals. Retrieved from https://reactjs.org/docs/faq-internals.html.
[7] Web Technology Surveys. Historical trends in the usage statistics of client-side programming languages for websites. Retrieved from https://w3techs.com/technologies/history_overview/client_side_language/all.
[8] React. State and Lifecycle. Retrieved from https://reactjs.org/docs/state-and-lifecycle.html

©IJRASET: All Rights are Reserved



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)