Umesh Agarwal, 2024, 12:3
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology

An Open Access Journal

React JS: A Open Source Modern Web Development Java Script Library for Front-end Development

Scholar Umesh Agarwal, Professor Dr. Vishal Shrivastava, Assistant Professor Dr. Akhil Pandey
Computer Science & Engineering
Arya College of Engineering & I.T. India, Jaipur

Abstract- React, also known as React.js, is a highly popular open-source JavaScript library used for building user interfaces in single-page applications and mobile app development. It's favored for its speed, simplicity, and scalability, offering features like JSX, stateful components, and a virtual Document Object Model (Virtual DOM). JSX is optional but aids in providing helpful error messages. Companies like Facebook and Instagram use React for large and complex online applications, enabling data updates without page refreshes for fast and robust web apps. React is versatile, supporting both simple and complex applications and can be used in the browser and on the server with Node.js. Why choose React over other front-end technologies? React offers a reliable choice with a vibrant community of over 1,400 developers and more than 92,000 websites. Its advantages include a smart diffing algorithm for efficient DOM updates, reusable components for easy app development, and a non-prescriptive approach, allowing a rich library selection. React's lifecycle methods and React Hooks provide a powerful way to manage component events throughout their lifecycle.

Keywords- React, prop types, react DOM, Hooks, Components, Virtual DOM.

I. INTRODUCTION

React, also known as React.js or React Js, is a powerful JavaScript library that excels in server-side rendering (SSR). It offers an efficient, flexible, and declarative approach to building reusable User Interface components for mobile and web applications. Created by Jordan Walke, a former Facebook software engineer, React.js has been widely used in various Facebook products like WhatsApp and Instagram. It was initially developed within Facebook's news feed section in 2011 but became public in May 2013.

ReactJS is known for its use of a virtual DOM-based mechanism to update the HTML DOM efficiently. The core of any React application lies in its components, each responsible for rendering a small, reusable piece of HTML code. In essence, React focuses on the view layer of mobile and web applications and is primarily concerned with rendering data to the Document Object Model (DOM).

React is a well-known open-source front-end JavaScript library, primarily created by Facebook. It has gained popularity within developer communities due to its simplicity and effectiveness in developing reusable UI components that handle data updates over time.

Each React component manages its state and updates the UI accordingly. This component-based approach allows data to be easily passed to the application while keeping the state out of the DOM. It's important to note that React is not an MVC framework like Angular; instead, it serves as a foundational library, emphasizing a straightforward and modular code structure.

React's key strength lies in its ability to source data either from a parent component or contain it within the component itself, making it highly efficient for building user interfaces.

The purpose of this paper is to conduct an in-depth analysis of the ReactJS library based on JavaScript. It will cover fundamental concepts, characteristics, features, development processes, architecture, and dependencies, aiming to provide a comprehensive understanding of ReactJS.

React.js is an open-source frontend JavaScript library for developing UI components, and it is maintained by Facebook along with contributions from individual developers and companies. React can serve as a foundation for single-page or Android applications, although it primarily focuses on state management and rendering to the document object model. To build complete React applications, developers often need to incorporate additional libraries for routing and certain client-side functionalities.

React JS, designed at Facebook by Jordan Walke, a Software Engineer in 2011 and released to the public in May 2013, places a strong emphasis on performance analysis. It achieves this by breaking down the UI into reusable components, promoting efficient development. React.js aligns with the view component of the Model-View-Controller (MVC) model and implements a one-way data flow, simplifying data binding. With its use of a virtual DOM, React offers straightforward programming and quick execution.

1. JSX (JavaScript Syntax Extension)

JSX, short for JavaScript XML, is a syntax extension used primarily with React to efficiently construct and render user interfaces. It compiles into JavaScript, resulting in improved code quality compared to equivalent JavaScript-only code. JSX simplifies the process of manipulating the Document Object Model (DOM) and is compatible with a wide range of web browsers, enhancing website performance and development efficiency. This extension combines HTML and JavaScript, providing a user-friendly XML or HTML-like format for coding, especially beneficial for those with experience in both HTML and JavaScript.

Figure 1: Javascript XML extension.

For instance, you can define a variable "n" as "Home," and then create an element like this:
~~~ javascript
const n = "Home";
const element = <h1>Welcome to {n}</h1>;

```javascript
const n = "Home";
const element = <h1>Welcome to {n}</h1>;
~~~

Figure 2: Javascript component example.

2. React Components

Create wrapper components that handle their own state, and then assemble them to construct intricate user interfaces. By implementing the item's logic in JavaScript instead of using templates, you can seamlessly transmit comprehensive data within your application while keeping the state separate from the Document Object Model (DOM). React applications are composed of numerous components, each possessing its own logic and functionalities, all written in JavaScript, which ensures efficiency and reusability. The React Component Lifecycle Diagram is shown below,i.e,

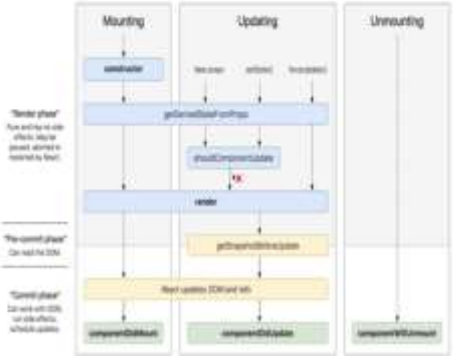


Figure 3: React component lifecycle.

3. One-way Data Binding or Data Flow

React utilizes one-way data binding to ensure that data flows in a singular direction, moving from parent components to child components. This one-directional data flow maintains the immutability of components, preventing any alterations to the data within them. Concepts like Flux have been designed to facilitate and maintain this unidirectional data flow, enhancing modularity and performance. React's architecture is structured to accommodate and support this downstream data flow, further reinforcing the unidirectional nature of data transfer throughout the application. Unidirectional flows offer greater control over the states and models within an application, simplifying its architecture and making it easier to comprehend. This approach also eliminates the need for cascading updates and modifications in the view layer[8].

4. Virtual DOM

The Virtual DOM, also known as the Document Object Model, serves as a representation of the original DOM object within a web application. It employs one-way data binding, whereby any changes made in the application first update the Virtual DOM. Subsequently, it identifies disparities between the real DOM and the Virtual DOM. Only the portions that have undergone changes are then updated in the real DOM. This approach results in enhanced application speed and reduced memory consumption. Certain DOM manipulations significantly improve web app performance, regardless of the client platform's speed and the JavaScript engine's efficiency, making them exceptionally valuable.

The DOM has a hierarchical tree structure, and modifications made at the top can have cascading effects on elements further down. This can lead to lag in user interface responsiveness and negatively impact the user experience. React.js addresses this issue by introducing a Virtual DOM layer between the user interface and the actual DOM, effectively mitigating these challenges. The Virtual DOM serves as a virtual representation of the DOM and remains in memory rather than being directly rendered on the screen. Any changes made to the Virtual DOM are managed by an algorithm, which determines the necessary adjustments to be made in the real DOM, consequently reflecting these changes on the user's screen. The Virtual DOM records and retains any modifications to the DOM in memory, and the algorithm specifies which elements need updating. This ensures that the user's screen is updated without disrupting other elements[7].

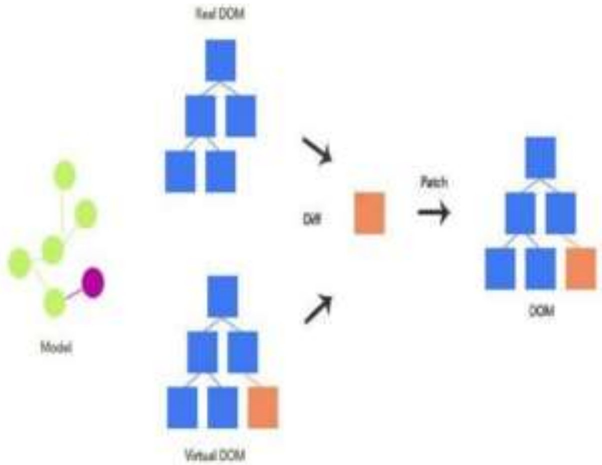


Figure 4: React Real and Virtual Dom working.

5. Extensions

React can be extended with various libraries and frameworks such as Flux, Redux, and React Native. These extensions enhance React's capabilities, allowing the creation of visually appealing user interfaces. React not only supports mobile app development but also provides server-side rendering. Numerous extensions are available for React to build comprehensive UI applications.

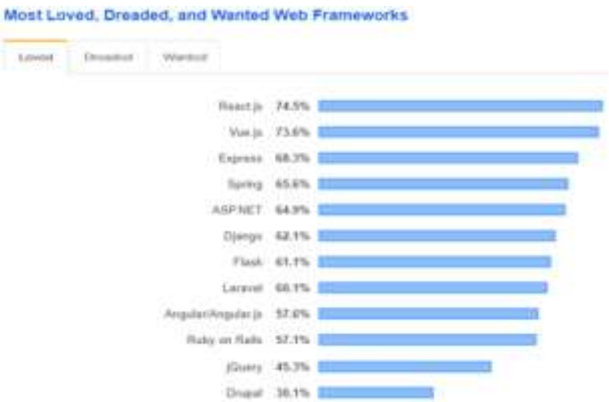


Figure 5: Popularity Chart

6. React VS Angular

The comparison between React & Angular is defined as [5] [6],

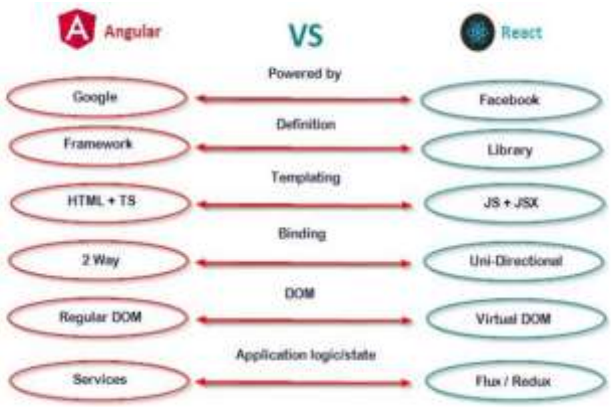


Figure 6: Angular v/s React comparision

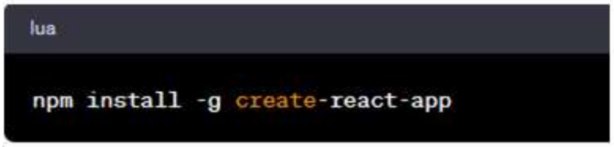
II. CREATING A REACT APPLICATION

React can be installed on various platforms, including macOS, Windows, and Unix. However, before installing React, it's essential to have Node and NPM preinstalled on your computer. There are two main methods for incorporating React into your application [2]:

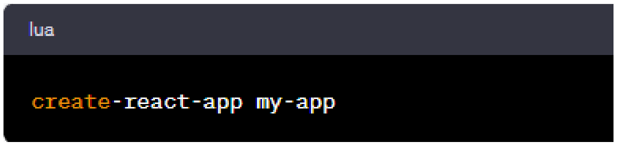
1. Automatic Installation of ReactJS

The automated process is used when creating a new application in React. Here are the steps:

Install the Create React App tool

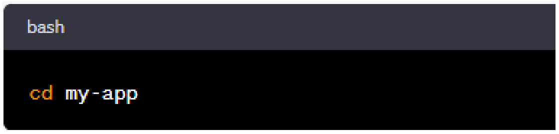
```

npm install -g create-react-app



~~~lua
npm install -g create-react-app
~~~

**Create a New React Application**

```

create-react-app my-app

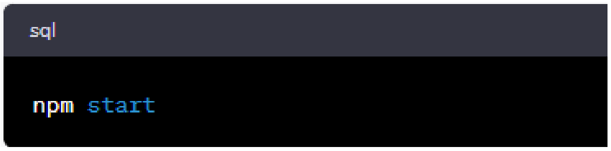


~~~lua
create-react-app my-app
~~~

Change to the Application's Directory

```

cd my-app



~~~bash
cd my-app
~~~

**Start the Development Server**

```

npm start



~~~sql
npm start
~~~

2. Manual Installation of ReactJS

The manual installation process is suitable when you want to use React in an existing application. You will need to install the following dependencies:

Requirements

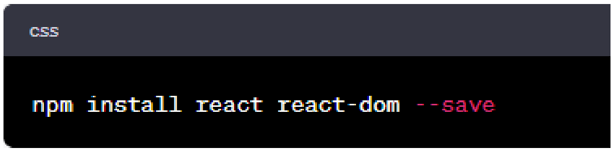
Web Pack

A bundler that enables you to write modular code and bundles all modules to enhance your application's performance.

Babel

A compiler that allows you to run modern JavaScript (like ES6) code in older browsers.

After installing Web pack and Babel, you need to install the React and React DOM modules with the following instructions:

```

npm install react react-dom –save



~~~css
npm install react react-dom --save
~~~

Once these installations are complete, you can create components and render them in your application.
```

III. LITERATURE SURVEY

React.js solves the problems of building UIs with 100% JavaScript by:

- Providing a declarative way to describe your UI, which makes it easier to track the dependency graph and avoid circular dependencies.
- Encapsulating your UI components in reusable modules, which makes it easier to maintain your code as your app scales.

Here is a reframed version of your sentences:

- Incremental markup building for deeply nested UI components requires tracking the dependency graph and avoiding circular dependencies in the rendering method.
- Modularity becomes difficult to maintain as app size grows.
- React.js solves these problems by providing a declarative way to describe the UI and encapsulating UI components in reusable modules [1].

IV. FUTURE SCOPE

ReactJS stands out as an easily accessible and widely favoured JavaScript framework. Businesses are increasingly adopting React due to its user-friendly nature and simplicity.

One of React's notable advantages is its ease of learning, making it stand out in comparison to other popular front-end frameworks like Angular and Vue. React's continued dominance is evident in its status as the number one web framework, according to Stack Overflow, and it maintains a strong position compared to competitors like jQuery and Vue.

This trend is not limited to the United States but extends to developing countries like India, where there has been a substantial 184% increase in open positions for React JS programmers post-COVID. As a result, React is poised to maintain its prominence on a global scale for years to come.

V. CONCLUSION

React JS has emerged as a valuable tool for developers, enabling them to create engaging web applications and user interfaces quickly. It simplifies component breakdown and facilitates the development of single-page applications with minimal coding. React is also recognized for its SEO-friendliness and suitability for building large-scale apps with frequently changing data, contributing to its widespread adoption.

React, a JavaScript-based front-end framework, offers substantial benefits to both developers and businesses. Its rapid rise in popularity is a testament to its advantages, such as robustness, responsiveness, and user-friendliness. Learning React is increasingly essential for developers, as it streamlines the creation of engaging web applications and large-scale data-driven projects.

React's strengths, including its advanced features, responsiveness, and user-friendliness, far outweigh its limitations. Organizations are actively promoting React's adoption, recognizing its market relevance. React's compatibility with various JavaScript features and its Virtual DOM feature improve overall application efficiency.

React's growth in demand is expected to continue, making it a crucial tool for companies aiming to achieve their goals. It was originally developed by Facebook for internal use and later open-sourced in 2013. Despite a few limitations, React has revolutionized web application development and is well-suited to meet modern requirements. React accelerates application adaptability and has the potential to significantly impact contemporary web development.

This paper explores the significance of React JS within the developer community and organizations. React's rapid adoption since its open-source release highlights its revolutionary impact on web development. The paper examines how React is a primary choice for frontend technologies and web application modeling. When compared to Angular, React is often the preferred solution due to its benefits, including working with a virtual DOM, simple UI design, JSX support, and a component-based architecture, making it a viable choice for both startups and enterprises.

In summary, React.js is a leading front-end library, known for its flexibility, performance, and strong community support. It's a future-proof option in the ever-changing landscape of front-end development.

REFERENCES

1. Alok Kumar Srivastava , Vaishnavi Laxmi , Payal Singh , Km Pratima , Vibha Kirti Assistant Professor, Department of Computer Science & Engineering. ("React JS (Open Source JavaScript Library)")Department of Information Technology, Buddha Institute of Technology, GIDA, Gorakhpur. FEB 2022.
2. Prateek Rawat , Archana N. Mahajan Student, Department of Computer Science and Engineering, Institute of Technology and Management Universe, Vadodara, Gujarat. Assistant Professor, Department of Computer Science and Engineering, ("ReactJS: A Modern Web Development Framework") Institute of Technology and Management Universe, Vadodara, Gujarat. NOV 2020.
3. Avinash Mishra, Arshita Gupta, Department Of Computer Science And Engineering, Malout Institute Of Management And Information Technology, Malout, Punjab, India. NOV 2022("REACT JS – AN FRONTEND JAVASCRIPT LIBRARY").
4. React.js and Front-End Development - DZoneWeb Dev.
5. Angular vs. React - A comparison - Haufe-Lexware.github.io (haufegroup.io).
6. Angular vs ReactJS : Which one is Most In-Demand Frontend Development Framework in 2019? – GeeksforGeeks.
7. React Front-End Development - 6 Things To Consider (brainhub.eu).
8. React (JavaScript library - Wikipedia).