

# Angular and React Comparison

Sagar Verma, Student, Department of MCA (Engg.), PES MCOE, Pune, India

Ass. Prof. Chandana Oak, Department of MCA (Engg.), PES MCOE, Pune, India

**Abstract**—In today's modern world every moment new trends and technologies surfaces, the field of web development seems to be in constant flux. To design and develop a website, frameworks have become an integral part. Frameworks provide ways to quickly develop and deploy interactive web pages. Frameworks are more adaptable for designing the websites most of the developers use it. As we all know JavaScript is the most popular programming language, the two of the most popular and trending frameworks in the tech arena are, Angular and ReactJS. Both of these frameworks are used for front end development. These two technologies are continually growing and are giving tough competition to each other and these two frameworks are maintained by biggest tech rivals, Google (Angular) and Facebook (ReactJS). ReactJS and Angular both are widely adopted JavaScript Framework used to develop interactive Single Page Applications (SPAs). In This Paper, we'll take a deep dive comparing JavaScript "frameworks," Angular and React.

**Keywords:** - JavaScript, Angular, ReactJS, Frameworks

## 1. Introduction

Companies have evolved over the last few years. A few years back focus was on entering the market with the best features, but now things have changed now it's more about the user experience. A good application in today's digital age is a blend of the visual frontend, robust backend and rich functionalities. Technology has become the backbone of any good application. There are multiple technologies that companies prefer for frontend and backend development, but only a few are well-received and popular. JavaScript is one such popular programming language. It is a cross-platform, object-oriented scripting language.

Some of the advantages of JavaScript include:

- Client-side execution.
- Easy to learn.
- Instant response to the visitors as the script is executed on the user's machine.
- Enhances interactivity.
- Rich interfaces which includes drag and drop components, sliders and much more
- Amazing community support.
- Provides frameworks for both ends, that is frontend and backend development.

We're living in the era of the JavaScript, as the language has quickly climbed to the first rank in the most used programming language, and is continuously gaining boost across different applications, not just front-end application development. These days, JavaScript is being used in the frontend, the backend, building mobile applications and even for building AI applications.

JavaScript was initially used only for the client-side. However, nowadays, JavaScript is used as a server-side programming language as well. To summarize, JavaScript is the language of the web. JavaScript frameworks are a collection of JavaScript libraries that provides developers with pre-written JavaScript code to use for building websites or web applications. A framework is a concept in which general functionality can be changed by additional user-written code. JavaScript framework is written in JavaScript where the programmers can make changes to the functions and use them for their convenience. Frameworks are more versatile for the designing of websites, and hence, most of the website developers prefer it. JavaScript frameworks help to work with JavaScript easier and smoother. These frameworks help the programmer to code the application as a device responsive. This responsiveness is yet another reason why the JavaScript frameworks are popular. Some of the top JavaScript Frameworks are.

- Angular
- React
- Express.js
- Node.js
- Vue.js
- Ember
- Meteor
- Polymer
- Backbone
- Redux

Now we are going to compare two of the most popular and trending frameworks of JavaScript and those are, Angular and React.

## 2. JavaScript Frameworks

JavaScript Frameworks we are going to compare are Single Page Applications. A single-page application (SPA) is a website design or web application approach through which each new page's content is presented not from loading new HTML pages but generated dynamically through JavaScript's ability to handle the DOM elements on the existing page itself. In a traditional web page application, an index.html page might be linked to other HTML pages on the server that the browser will download and display from scratch. A SPA approach allows the user to continue interacting and gathering information from the page while new elements are being updated or fetched, and can result in much faster interactions and content reloading. Also, the SPA's allows changing the page's URL without reloading the page, which in turn helps to create different URLs for different views. [10]

Facebook, Gmail, and GitHub are popular examples of Single-Page Applications (SPAs). A SPA removes the reloading of the web page when in use by loading all the content through JavaScript. SPAs work inside the browser and show natural behaviour. Users like it because SPAs do not require additional waiting time, and directly render the web pages in the user's browser.

Pros of the Single-Page Application:

- SPA is fast, as most resources are only loaded once throughout the life of the application. Only data is transmitted back and forth.
- The development is simplified and streamlined. There is no need to write code to display pages on the server. It is much easier to get started because you can start development from a file, without using any server at all.
- SPAs are easy to debug, as you can check network operations, examine page elements and data associated with it.
- It's easier to convert it into a mobile application because the developer can reuse the same backend code for web application and native mobile application.
- SPA can cache any local storage effectively. An application sends only one request, and it stores all the data, then it can use the stored data and the SPA even works offline.

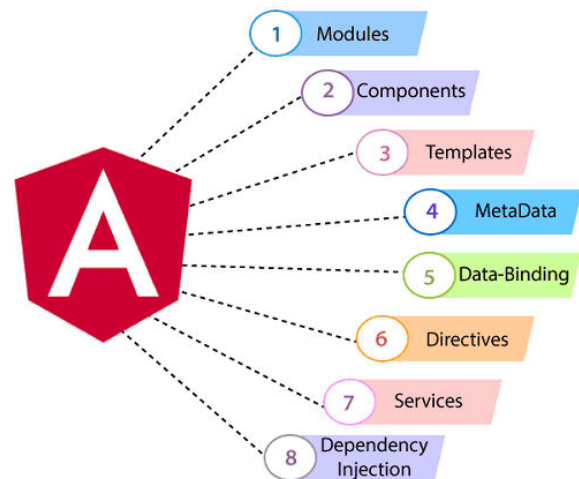
Single Page Applications are a great tool for making incredibly engaging and unique experiences for users. Single page application is a single page (hence the name) where a lot of information remains the

same and only a few things need to be updated at a time.

### A. Angular

Angular is an open-source Web framework, developed by an Angular team at Google and maintained by the tech giant (Google) itself and in no time from its launch in 2010, it has become quite popular among development communities. Angular expands the use of features of HTML and can be used to launch applications faster than any other framework. The framework is the part of MEAN stack and which is mainly used for developing SPAs (Single Page Applications). It is based on the TypeScript environment; TypeScript is a superset of JavaScript that provides static typing and a few compelling features that are not present in the regular JavaScript. Angular is a platform and a framework used to create web applications in HTML and TypeScript. Angular is written in TypeScript. It helps implement core and optional functionality through libraries that you can import into your apps. [7]

#### Key parts of Angular

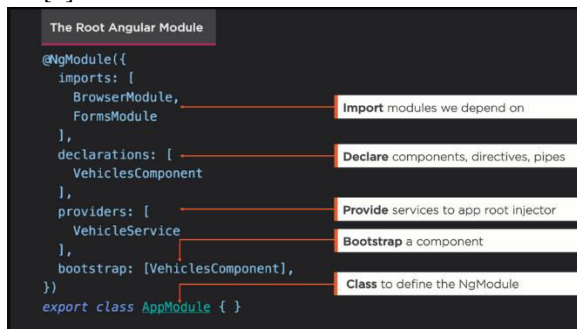


#### 1. Modules

In Angular, a module is used to group components, directives, pipes and services together, in such a way that all can be integrated with the other modules to create an application. Every Angular app has at least one NgModule class, the root module, which is formally named AppModule and exists in a file named app.module.ts.

A NgModule is defined by a class marked with @NgModule(). The @NgModule() decorator is a function that features a single metadata object, whose properties describe the module. The most important properties are as follows.

- declarations: Includes components, directives, and pipes that belong to NgModule.
- imports:Includes other modules whose exported classes are needed by component templates declared in NgModule.
- providers:Include services that NgModule contributes to the global collection of services; they become accessible in all parts of the app.
- bootstrap: Includes main application view, called the root component, which hosts all other app views. Only the root NgModule has the bootstrap property set.[1]



```

The Root Angular Module
@NgModule({
  imports: [
    BrowserModule,
    FormsModule
  ],
  declarations: [
    VehiclesComponent
  ],
  providers: [
    VehicleService
  ],
  bootstrap: [VehiclesComponent],
})
export class AppModule { }

```

## 2. Components

An Angular application consists of various components with a root component. A component controls the view and behaviour of the web application. It contains TypeScript, HTML and CSS code.

Each component consists of 4 files

- component.html: This file consists of the HTML code for how the component will view and look.
- component.spec.ts: This file is in typescript which is used for testing purpose. The presence of this file is not necessary.
- component.ts: This file consists of the component class in typescript format with the component name which implements the OnInit interface and has a ngOnInit() method by default. The code written within this file is executed when the page life-cycle state is Initialized. The TypeScript file contains all the code used by the component to control its behaviour.
- component.css: This file has the CSS which is linked to the HTML file and contains all the CSS styles for the component.

A component must be included in declarations array in NgModule for it to be available to another component or application. To make it a member of a NgModule, list it in the declarations array of the NgModule.

## 3. Template

A template is an HTML part that tells Angular how to display the component in angular application. The template is associated with a component which defines the component’s view. The template is an important part of an Angular component and is necessary for the component to work. In Angular, the component acts as the controller or view model, while the template represents the view itself.

## 4. MetaData

Metadata is used to provide data about the application to the class so that it can know the expected behaviour or logic of the Application. Decorators are the root concept when developing with Angular. The developer can use metadata to a class to tell Angular app that AppComponent is the component. Metadata can be added to the TypeScript using the decorator.

```

// app.component.ts
@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})

```

@Component is a decorator which uses configurationobject to name the component and create the component and its view.

## 5.Data Binding

Angular allows communication between a component and DOM, making it easy to create an interactive application without worrying about sending and receiving data.[8]

From the Component to the DOM

Interpolation: {{value}}: Interpolation adds the value of the property from the component.

```
<p>Name: {{student.name}} </p>
```

```
<p>College: {{student.College}}</p>
```

Property binding: [property]="value"

In property binding, a value is passed from a component to a specified property, which can often be a simple html attribute.

```
<input type="text" [value]="student.name" />
```

```
<input type="text" [value]="student.college" />
```

## 6. Directives

The Angular directives are used to manipulate the DOM. By using Angular directives, you can change the appearance, behaviour or a layout of a DOM element.

There are three kinds of directives in Angular:

- Components are the most common directives. Component directives are used in the main class. They contain the detail of the component, it’s a

directive with a template. The component directive is used to specify the HTML templates. It has structure design and the working pattern of the component.

- Structural Directives are used to change the structure of the view. Structural directives start with a \* sign. Structural directives use the DOM layout to add and remove DOM to change the view. For example, \*ngIf directive, \*ngSwitch directive, and \*ngFor directive.
- Attribute directives are used as attributes of HTML elements. Attribute directives change the appearance or behaviour of an HTML element, component, or another directive. For example, ngClass directive, and ngStyle directive etc.

### 7.Dependency Injection

Dependency injection (DI) helps to keep component classes lean and efficient. DI does not fetch data from a server or validate the user input, or log directly to the console instead they pass on such tasks to the services. DI is injected into an Angular framework and used everywhere to provide new components with the services or other things they need. Components use services; that is, you can inject a service into a component, giving the component access to that service class.[9]

### 8.Services

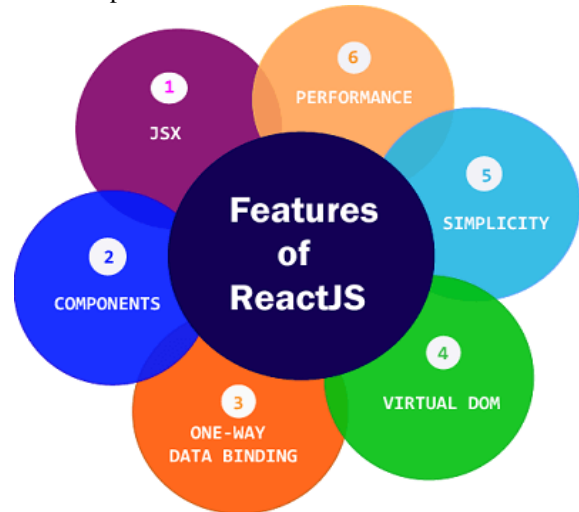
For data or logic that isn't related to a specific view, and that is to be shared between components, a service class is created. The @Injectable decorator is declared before the service class definition. The decorator provides the metadata that allows service to be injected into components as a dependency. Angular differentiate components from services to increase modularity and reusability. By separating a component's view-related functionality from other kinds of processing, such as data and logic, it makes component classes lean and efficient.



## B. React

ReactJS is an open-source JavaScript library that's used for building user interfaces specifically for single-page applications. It's used for handling the view layer of web and mobile apps. React also allows us to make reusable UI components. React allows developers to make large web applications which will change data, without reloading the page. It is maintained by Facebook and employed by many companies today for UI development. React JS is

additionally called simply to React or React.js. Some of the important features of ReactJS are as follows:



### 1.JSX

JSX stands for JavaScript XML. It allows us to write HTML to React and makes it easier to write and add HTML to React. It also allows us to write HTML elements in JavaScript and place them in the DOM. JSX converts HTML tags into react elements. JSX is an XML/HTML-like syntax used by React that extends ECMAScript so that XML/HTML-like text can co-exist with JavaScript/React code. The syntax is set to be used by pre-processors (i.e., transpilers like Babel) to transform HTML-like text found in JavaScript files into standard JavaScript objects that a JavaScript engine will parse. Basically, by using JSX one can write concise HTML/XML-like structures (e.g., DOM like tree structures) in the same file as we write JavaScript code, then Babel will transform these expressions into actual JavaScript code.[2]

### 2.Components

ReactJS is all about components. ReactJS application consists of more than one component, and each component has its logic and behaviour. These components can be reusable which help to maintain the code when working on larger applications. Components are the building blocks of any ReactJS app and a typical app will have many of these. In Simplest Terms, a component is a JavaScript class or function that accepts inputs that are properties(props) and in return, it gives a React element that describes how an area of the UI (User Interface) should appear. Components are of two types- Class components and Function components. When creating a React component, the component name must start with an upper-case letter. The component has to include the extends ReactComponent statement, this statement creates an inheritance to ReactComponent and gives

your component access to ReactComponent functions

### 3. One-Way Data Binding

ReactJS is constructed in such a fashion that it follows unidirectional data flow or one-way data binding. The benefits of one-way data binding give better control throughout the application. If the data flow is in the opposite direction, then it requires additional features. It is because components are supposed to be unchangeable and the data within them cannot be changed. React apps are formulated as a series of nested components. These nested components are functional: meaning, they obtain data through arguments (which is in props attribute) and pass that data through their return values (the return value of the render function). This method is called unidirectional data flow or one-way data binding, in which data is passed down from components to their children components.

### 4. Virtual DOM

DOM stands for “Document Object Model”. The DOM in simple words represents the UI of an application. Every time there is a change in the state of an application UI, the DOM gets updated to represent that change. A virtual DOM object is a depiction of a DOM object. It is like a lightweight copy which is detached from the browser-specific implementation details. In simple language, the Virtual DOM is React’s the local copy of the HTML DOM. It allows React to do its data processing and skip the real DOM operations. When any of the virtual DOM object changes, then React updates those objects, and only those objects, on the real DOM. [2]

### 5. Simplicity

ReactJS uses JSX file which makes the application simple to code as well as understands. We know that ReactJS uses a component-based approach which makes the code reusable as per the need. This makes it simple to use and learn. The functional approach to rendering HTML. In React.js, we can know which given component will render based on the properties it is supplied with. It’s easy to know exactly what will happen. That directness and simplicity help greatly with building software. It’s the big appeal of functional languages like ReactJS. What makes React just so good? It’s simple, It’s automated, and It’s open.

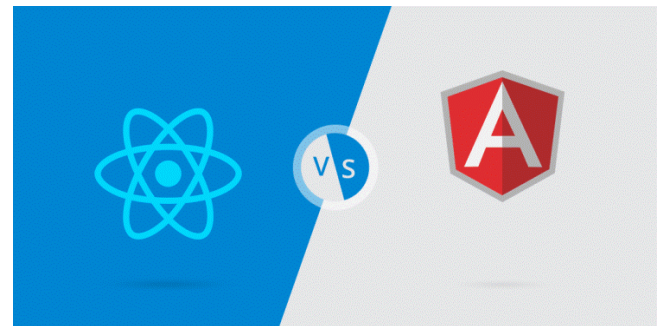
### 6. Performance

ReactJS is known as a great performer. This feature makes it much better than other frameworks out there. The reason behind this is that it has a virtual

which is the root component.

DOM. The DOM is a cross-platform and programming API which handles HTML, XML or XHTML. The DOM exists entirely in memory. Due to this, when a component is created, one did not write directly to the DOM. Instead, one will write virtual components that will turn into the DOM leading to smoother and faster performance. ReactJS guarantees code stability and better app performance by using only unidirectional data flow. When it comes to changing an object, the developers only modify its state and make changes thereby some specific components are updated which then allow better stability and efficient app performance.

### C. Comparison



Most prominent features of Angular:

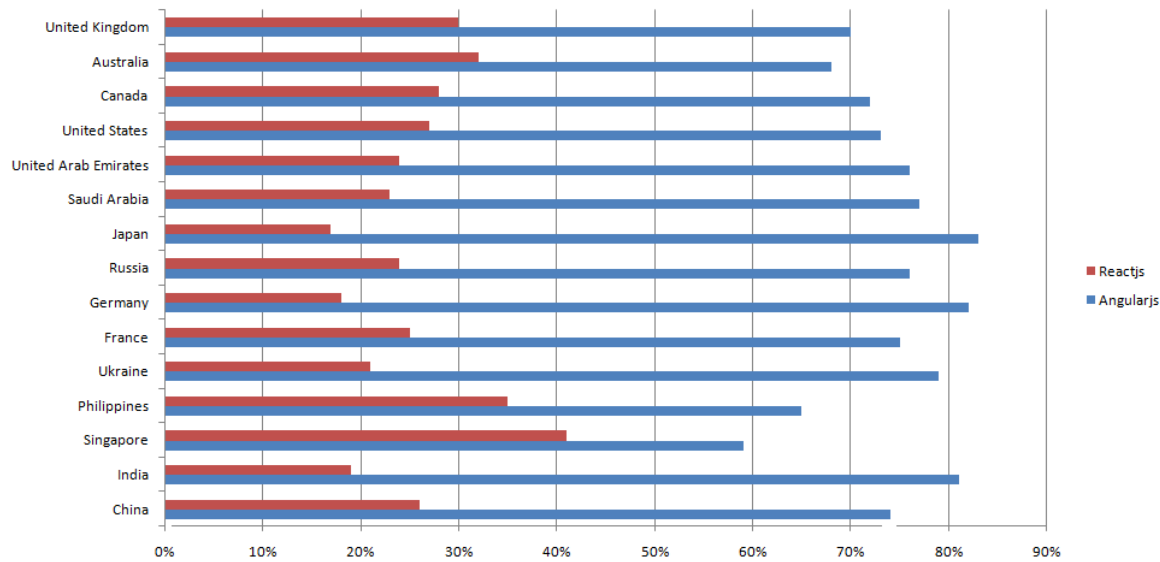
1. It is well supported by the MVC architecture making development and testing quicker thereby improving the productivity of development teams.
2. A framework that is designed for web developers to have better control.
3. Features like dependency injection and data-binding help to reduce the time for writing the codes.
4. Piping, is one other feature, a predefined keyword which is used as a symbol. Before presenting the data to users, an angular pipe modifies the data. These pipes can be used with directives and expressions.
5. By using expressions, Angular can directly manage the data within the templates.
6. The framework is built keeping CRUD (create, read, update and delete) concept in mind.

Following are the top features of ReactJS that makes it unique in its way:

1. ReactJS comes with a concept of component-based architecture.
2. It supports languages like JSX.
3. Availability of ECMAScript 6 and 7
4. Easy interface design and learning of API
5. Excellent for JavaScript debugging.

<b>Characteristic</b>	<b>ReactJS</b>	<b>Angular</b>
Creator	Facebook	Google
Language	JSX	JavaScript or HTML
Version	v16.13.0	Angular 9.0.2
DOM	Virtual DOM	Regular DOM
Data- binding	Uni-Directional	Two-way
Routing	React-Router	Own routing functionality
Speed	310 milliseconds	1.35 seconds
Templating	In JSX file	In HTML
Architecture	No Architecture	MVC architecture
Mobile Support	React Native	Ionic Framework
ECMA	Babel	Typescript

6. It renders a large amount of data. [6]



### The popularity of Angular vs ReactJS across International Region

ReactJS and Angular comparison is still a popular topic in 2020. ReactJS and Angular are both advanced and widely adopted JavaScript (JS) technologies that are used to create interactive single-page applications (SPAs). The number of JavaScript tools for developing a single-page application is constantly growing, choosing which technology to rely on and choose, is getting more challenging for web developers. Both ReactJS and Angular are currently used by many companies, news, and travel companies in the USA, the UK, Canada, France, Germany, Australia, and other countries.[4]

Brands using ReactJS

- Uber
- Dropbox
- Flipkart
- Expedia
- Airbnb
- Netflix
- IMDb
- PayPal
- Instagram

Brands using Angular

- PS3 YouTube App
- NASA
- HBO Official Site
- Nike
- General Motors
- AWS
- Open Stack
- iTunes Connect
- Gmail

Product development companies are leveraging various digital technologies. Their main goal is not only to provide a feature-rich product but also intuitive user experience. ReactJS and Angular could both help them to achieve this goal. While both are equally amazing and have a strong ecosystem, both offer a diverse approach to web application development.

### D. Conclusion

Both Angular and ReactJS are great frameworks for writing single-page applications. But they are completely different. Some programmers say that ReactJS is better than Angular or vice versa. What's best for a given project, however, depends on how one uses these frameworks. Working with ReactJS may seem a bit easier at starting because you write old-school JavaScript and build your HTML around it. There are plenty of things to compare between Angular and ReactJS, what matters the most is what fits the project. Both carry some pros and cons; however, it is the projects and business environment that plays a key role. One should think about their plans and then choose the development technology.

As we know that there are bundle of frameworks of JavaScript including Ember, Node, Polymer, Meteor, and more, but Angular and ReactJS are the real titans of the rapidly advancing ecosystem. I tried to compare the different parameters of each framework without getting biased. My only aim was to provide

and help you understand the background and functionality of these frameworks, so you can pick the right framework for your next web development project.[5]

### E. Future Scope

Facebook and ReactJS team have shown the commitment to enhancing ReactJS efficiency. This is crucial to whether the fast-emerging competition from other frameworks such as Angular.

Some of the things expected in future releases include;

ReactJS enthusiasts should expect new render types. For example, the ReactJS team is likely to add unique syntax to the JSX that does not need keys.

Developers are also expecting an improvement in error handling. In the past, cases of runtime errors during rendering often put ReactJS in an awkward state. This requires regular page refreshing. The current model of solving this error is by throwing it inside the component or lifecycle methods. Though this helps to stop rendering corrupted data though it is

not the best idea for better UX. The latest ReactJS comes with a rewritten server render which makes it very fast and ideal.

The development of the Angular framework has now attained very serious momentum. The angular team is trying to improve the framework in all directions.

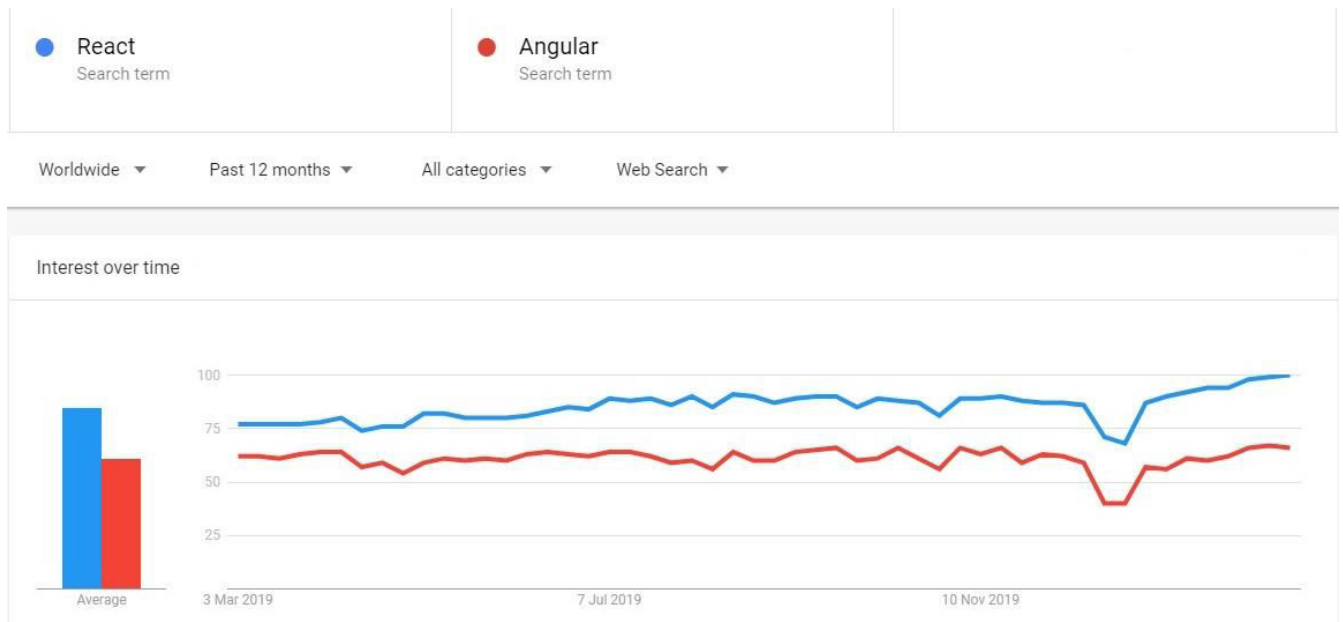
Though the main directions would include:

- Building optimization
- Compiler Improvements
- CLI

These points will help to upgrade Angular to be smaller, faster, and easier to use. Angular 8.0 was the first step for Ivy (Ivy is a complete rewrite of the compiler and runtime) & Angular 9.0 has released now, and Ivy has become the default rendering engine. And then the team is working on adding other features such as to lazy-load a single component instead of a module. Or to have JIT components and AOT components work with each other. Or to manually write a template, by hand-writing the generated instructions to squeeze the best performances.[3]

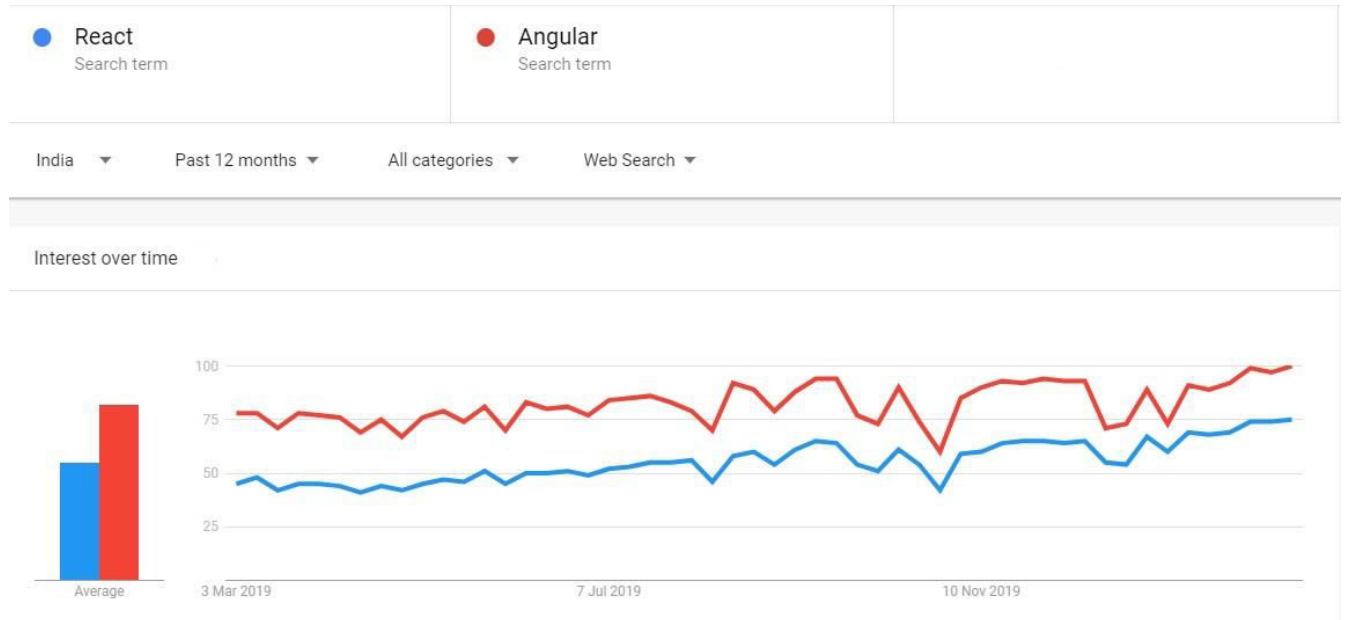
### F. Trends

#### 1. Trend Worldwide





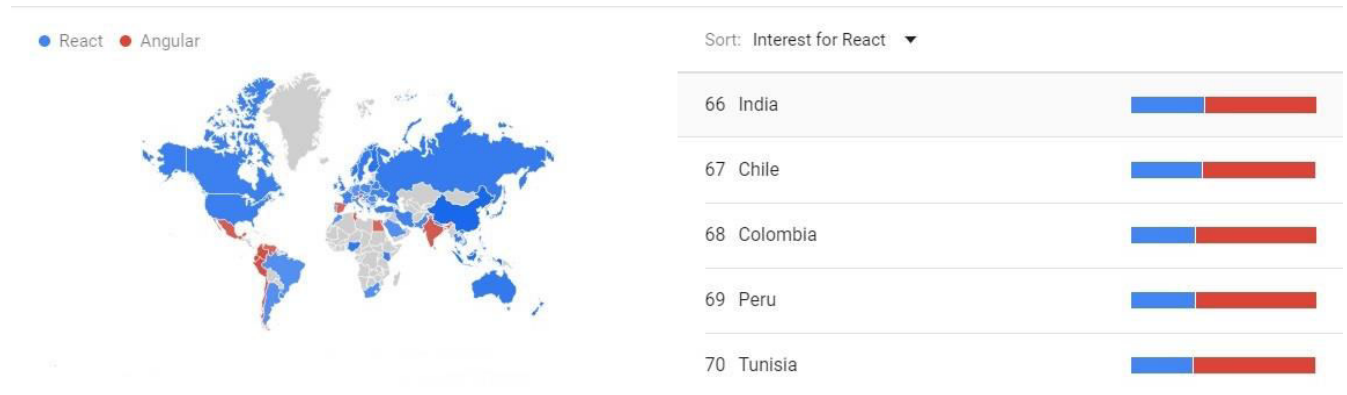
## 2.Trend India



## 3.Trend for ANGULAR (Regional)



#### 4. Trend for REACT (Regional)



#### G. References

- [1] <https://angular.io/guide/architecture-modules>
- [2] <https://reactjs.org/docs/introducing-jsx.html>
- [3] <https://medium.com/better-programming/angular-in-2020-and-beyond-b2e98543ef17>
- [4] <https://rubygarage.org/blog/react-vs-angularjs>
- [5] <https://hackr.io/blog/angular-vs-react>
- [6] <https://www.tothenew.com/blog/angularjs-vs-reactjs-comparison/>
- [7] Sneha Ambulkar “Angular JS”, IJSER, Vol. 7, 2016
- [8] Nilesh Jain, Priyanka Mangal and Deepak Mehta “AngularJS: A Modern MVC Framework in JavaScript”, JGRCS Volume 5, No. 12, December 2014
- [9] Preeti Yadav, Bhupender Singh “A comparative study of versions of JavaScript”, IJCIR, Vol. 13, 2017
- [10] Anurag Kumar, Ravi Kumar Singh “Comparative analysis of Angular JS and React JS”, International Journal of Latest Trends in Engineering and Technology, Vol. 7, 2017