EXPERIMENT 4a

NAME: Shivam Pawar **UID: 2019230068**

NAME: Vishal Salvi **UID: 2019230069**

NAME: Shreyas Patel **UID: 2018130043**

CLASS: TE COMPS **BATCH: C**

DATE:

Aim: Understanding and use of Nodejs, Angularjs, Reactjs etc.

Theory:

Introduction to Node.js

The modern web application has really come a long way over the years with the introduction of many popular frameworks such as bootstrap, Angular JS, etc. All of these frameworks are based on the popular JavaScript framework.

But when it came to developing server-based applications, there was a kind of void, and this is where Node.js came into the picture.

Node.js is also based on the JavaScript framework, but it is used for developing server-based applications. While going through the entire tutorial, we will look into Node.js in detail and how we can use it to develop server-based applications.

What is Node.js?

Node.js is an open-source, cross-platform runtime environment used for the development of server-side web applications. Node.js applications are written in JavaScript and can be run on a wide variety of operating systems.

Node.js is based on an event-driven architecture and a non-blocking Input/Output API that is designed to optimize an application's throughput and scalability for real-time web applications.

Over a long period of time, the framework available for web development were all based on a stateless model. A stateless model is where the data generated in one session (such as information about user settings and events that occurred) is not maintained for usage in the next session with that user.

A lot of work had to be done to maintain the session information between requests for a user. But with Node.js, there is finally a way for web applications to have real-time two-way connections, where both the client and server can initiate communication, allowing them to exchange data freely.

Why use Node.js?

We will have a look into the real worth of Node.js in the coming chapters, but what is it that makes this framework so famous. Over the years, most of the applications were based on a stateless request-response framework. In these sorts of applications, it is up to the developer to ensure the right code was put in place to ensure the state of web session was maintained while the user was working with the system.

But with Node.js web applications, you can now work in real-time and have a 2-way communication. The state is maintained, and either the client or server can start the communication.

Features of Node.js

Let's look at some of the key features of Node.js

1. Asynchronous event-driven IO helps concurrent request handling – This is probably the most significant selling point of Node.js. This feature basically means that if a request is received by Node for some Input/Output operation, it will execute the operation in the background and continue with processing other requests.

2. Node uses the V8 JavaScript Runtime engine, the one which is used by Google Chrome. Node has a wrapper over the JavaScript engine which makes the runtime engine much faster and hence the processing of requests within Node also become faster.
3. Handling of concurrent requests – Another key functionality of Node is the ability to handle concurrent connections with a very minimal overhead on a single process.
4. The Node.js library uses JavaScript – This is another important aspect of development in Node.js. A major part of the development community is already well versed in javascript, and hence, development in Node.js becomes easier for a developer who knows javascript.
5. There is an active and vibrant community for the Node.js framework. Because of the active community, there are always keys updates made available to the framework. This helps to keep the framework always up-to-date with the latest trends in web development.

Who uses Node.js?

Node.js is used by many large companies. Below is a list of a few of them.

- PayPal – A lot of sites within Paypal have also started the transition onto Node.js.
- LinkedIn - LinkedIn is using Node.js to power their Mobile Servers, which powers the iPhone, Android, and Mobile Web products.
- Mozilla has implemented Node.js to support browser APIs which has half a billion installs.
- eBay hosts their HTTP API service in Node.js

When to Use Node.js

Node.js is best for usage in streaming or event-based real-time applications like

1. Chat applications
2. Game servers – Fast and high-performance servers that need to processes thousands of requests at a time, then this is an ideal framework.
3. Good for collaborative environment – This is good for environments which manage documents. In a document management environment, you will have multiple people who post their documents and do constant changes by checking out and checking in documents. So Node.js is good for these environments because the event loop in Node.js can be triggered whenever documents are changed in a document managed environment.
4. Advertisement servers – Again here you could have thousands of requests to pull advertisements from the central server and Node.js can be an ideal framework to handle this.
5. Streaming servers – Another ideal scenario to use Node is for multimedia streaming servers wherein clients have request's to pull different multimedia contents from this server.

Node.js is good when you need high levels of concurrency but less amount of dedicated CPU time.

Best of all, since Node.js is built on javascript, it's best suited when you build client-side applications which are based on the same javascript framework.

When to not use Node.js

Node.js can be used for a lot of applications with various purposes. The only scenario where it should not be used is where there are long processing times, which is required by the application.

Node is structured to be single-threaded. If an application is required to carry out some long-running calculations in the background, it won't be able to process any other requests. As discussed above, Node.js is used best where processing needs less dedicated CPU time.

AngularJS

AngularJS is a Javascript framework used for building MVC based applications. This course is a step by step guide to learn AngularJS components like directives, filters, expressions, etc. and explore more of its programming aspect for SPA's.

What is AngularJS?

AngularJS is an open source Model-View-Controller framework which is similar to the JavaScript framework.

Angular JS is probably one of the most popular modern-day web frameworks available today. This framework is used for developing mostly Single Page applications. This framework has been developed by a group of developers from Google itself.

Because of the sheer support of Google and ideas from a wide community forum, the framework is always kept up to date. Also, it always incorporates the latest development trends in the market.

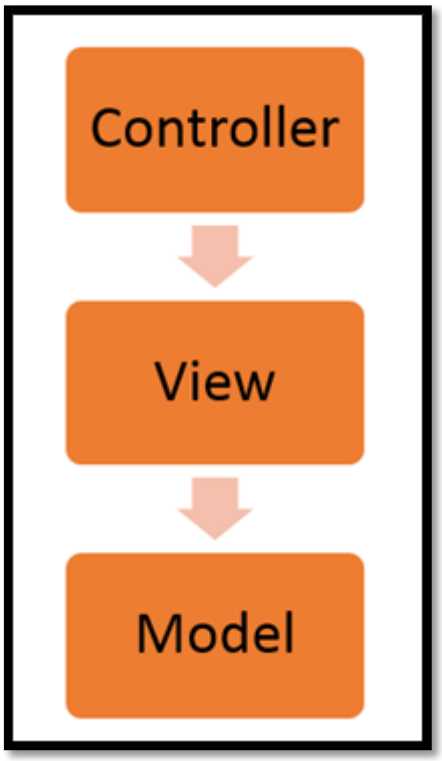
AngularJS Features

Angular has the following key features which makes it one of the powerful frameworks in the market.

1. **MVC** – The framework is built on the famous concept of MVC (Model-View-Controller). This is a design pattern used in all modern-day web applications. This pattern is based on splitting the business logic layer, the data layer, and presentation layer into separate sections. The division into different sections is done so that each one could be managed more easily.
2. **Data Model Binding** – You don't need to write special code to bind data to the HTML controls. This can be done by Angular by just adding a few snippets of code.
3. **Writing less code** – When carrying out DOM manipulation a lot of JavaScript was required to be written to design any application. But with Angular, you will be amazed with the lesser amount of code you need to write for DOM manipulation.
4. **Unit** Testing ready – The designers at Google not only developed Angular but also developed a testing framework called "Karma" which helps in designing unit tests for AngularJS applications.
5. Built-in support for AJAX, HTTP, and Observables
6. Large community support
7. Consistent with technology
8. Typescript offers efficiency
9. Cleaner and crisp Coding
10. Enhanced support for error handling
11. Seamless updates using Angular CLI
12. Forms and validation
13. Shadow DOM / local CSS
14. UI and Business Logic Separation

AngularJS Architecture

Angular.js follows the MVC architecture, the diagram of the MVC framework as shown below.

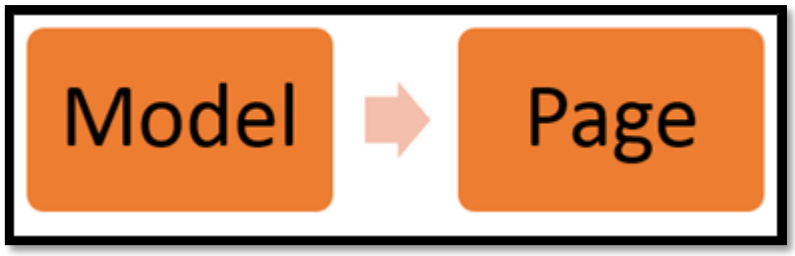


Angularjs Architecture Diagram

- The Controller represents the layer that has the business logic. User events trigger the functions which are stored inside your controller. The user events are part of the controller.
- Views are used to represent the presentation layer which is provided to the end users
- Models are used to represent your data. The data in your model can be as simple as just having primitive declarations. For example, if you are maintaining a student application, your data model could just have a student id and a name. Or it can also be complex by having a structured data model. If you are maintaining a car ownership application, you can have structures to define the vehicle itself in terms of its engine capacity, seating capacity, etc.

AngularJS Advantages

- Since it's an open source framework, you can expect the number of errors or issues to be minimal.
- Two-way binding – Angular.js keeps the data and presentation layer in sync. Now you don't need to write additional JavaScript code to keep the data in your HTML code and your data later in sync. Angular.js will automatically do this for you. You just need to specify which control is bound to which part of your model.



- Routing – Angular can take care of routing which means moving from one view to another. This is the key fundamental of single page applications; wherein you can move to different functionalities in your web application based on user interaction but still stay on the same page.
- Angular supports testing, both Unit Testing, and Integration Testing.
- It extends HTML by providing its own elements called directives. At a high level, directives are markers on a DOM element (such as an attribute, element name, and comment or CSS class) that tell AngularJS's HTML compiler to attach a specified behavior to that DOM element. These directives help in extending the functionality of existing HTML elements to give more power to your web application.

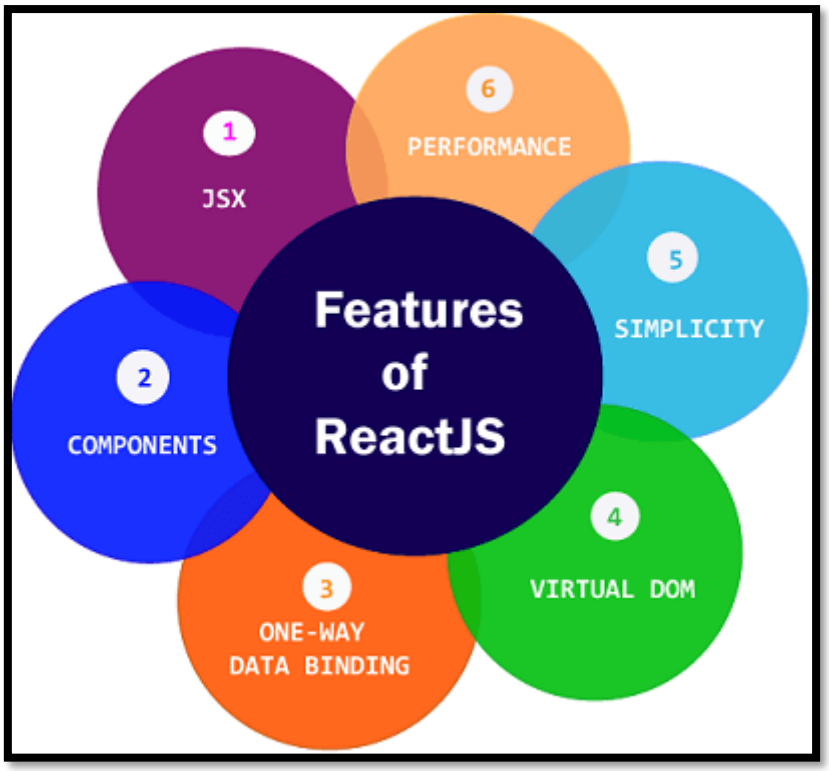
What is React JS?

REACT JS is an open-source front-end JavaScript library for building user interfaces. **ReactJS** is maintained by Facebook and a community of individual developers and companies. It is widely used as a base in building single-page websites and mobile applications. It is very easy to use, and it allows users to create reusable UI components.

Features of ReactJS

Currently, ReactJS gaining quick popularity as the best JavaScript framework among web developers. It is playing an essential role in the front-end ecosystem. The important features of ReactJS are as following.

- JSX
- Components
- One-way Data Binding
- Virtual DOM
- Simplicity
- Performance



JSX

JSX stands for JavaScript XML. It is a JavaScript syntax extension. Its an XML or HTML like syntax used by ReactJS. This syntax is processed into JavaScript calls of React Framework. It extends the ES6 so that HTML like text can co-exist with JavaScript react code. It is not necessary to use JSX, but it is recommended to use in ReactJS.

Components

ReactJS is all about components. ReactJS application is made up of multiple components, and each component has its own logic and controls. These components can be reusable which help you to maintain the code when working on larger scale projects.

One-way Data Binding

ReactJS is designed in such a manner that follows unidirectional data flow or one-way data binding. The benefits of one-way data binding give you better control throughout the application. If the data flow is in another direction, then it requires additional features. It is because components are supposed to be immutable and the data within them cannot be changed. Flux is a pattern that helps to keep your data unidirectional. This makes the application more flexible that leads to increase efficiency.

Virtual DOM

A virtual DOM object is a representation of the original DOM object. It works like a one-way data binding. Whenever any modifications happen in the web application, the entire UI is re-

rendered in virtual DOM representation. Then it checks the difference between the previous DOM representation and new DOM. Once it has done, the real DOM will update only the things that have actually changed. This makes the application faster, and there is no wastage of memory.

Simplicity

ReactJS uses JSX file which makes the application simple and to code as well as understand. We know that ReactJS is a component-based approach which makes the code reusable as your need. This makes it simple to use and learn.

Performance

ReactJS is known to be a great performer. This feature makes it much better than other frameworks out there today. The reason behind this is that it manages a virtual DOM. The DOM is a cross-platform and programming API which deals with HTML, XML or XHTML. The DOM exists entirely in memory. Due to this, when we create a component, we did not write directly to the DOM. Instead, we are writing virtual components that will turn into the DOM leading to smoother and faster performance.

Advantages of ReactJS

Here, are important pros/benefits of using ReactJS:

- ReactJS uses virtual dom that makes use of in-memory data-structure cache, and only the final changes are updated in browsers dom. This makes the app faster.
- You can create components of your choice by using the react component feature. The components can be reused and also helpful in code maintenance.
- Reactjs is an open-source javascript library, so it is easy to start with.
- ReactJS has become very popular in a short span and maintained by Facebook and Instagram. It is used by many famous companies like Apple, Netflix, etc.
- Facebook maintains ReactJS, the library, so it is well maintained and kept updated.
- ReactJS can be used to develop rich UI for both desktop and mobile apps.
- Easy to debug and test as most of the coding is done in javascript rather than on Html.

Disadvantages of ReactJS

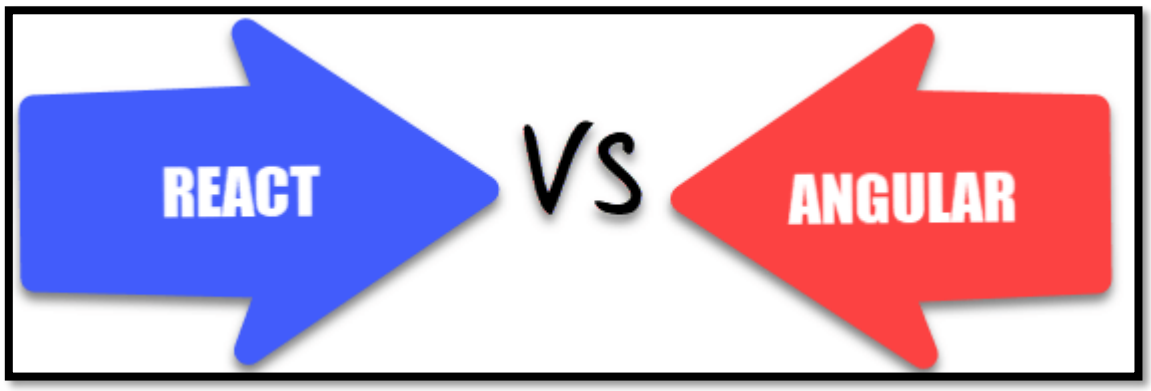
Here, are cons/ drawbacks of using ReactJS:

- Most of the code is written in JSX, i.e., Html and css are part of javascript, it can be quite confusing as most other frameworks prefer keeping Html separate from the javascript code.
- The file size of ReactJS is large.

Why we use ReactJS?

The main objective of ReactJS is to develop User Interfaces (UI) that improves the speed of the apps. It uses virtual DOM (JavaScript object), which improves the performance of the app. The JavaScript virtual DOM is faster than the regular DOM. We can use ReactJS on the client and server-side as well as with other frameworks. It uses component and data patterns that improve readability and helps to maintain larger apps.

React vs. Angular: The Complete Comparison



Parameters	React	Angular
Type	React is a JavaScript library, and it is much older compared with Angular.	Angular is a complete framework.
Use of libraries	React js can be packaged with other programming libraries.	Angular is a complete solution in itself.
Learning curve	It is easier to grasp compared Angular. However, it is difficult to learn when augmented with Redux.	Learning Angular is not easy for beginners. Thus, it requires lots of training.
Community support	When it comes to community support React doesn't offer much.	It has a viable and dependable community support system
Installation time	React takes longer to set up. But, it is really fast for delivering projects and building apps.	Angular is easy to set up but may lead to an increase in coding time which also results in delayed project deliveries.
Best feature	It gives you the freedom to choose the tools,	It offers a limited amount of freedom and flexibility.

	architecture, and libraries, for developing an app.	
Data binding	React language uses one-way data binding, which means that the Ul elements can't be changed without updating the corresponding model state.	Angular, on the other hand, uses the two-way data binding method. It helps you to ensure that the model state automatically changes when any change is made.
Testing & Debugging	It requires a set of tools to perform different types of testing.	The testing and debugging for a complete project is possible with a single tool.
Documentation	Although it is also undergoing regular updates, the documentation is relatively faster.	Due to the ongoing development process, the documentation is slower.
Updates	Updates in React are simple because scripts help in the migration.	It plans updates every six months, which gives some time to make needed changes for migration.
Application Types	Use this app if you want to develop Native apps, hybrid apps, or web apps	You should use this framework If you want to develop a SPA(Single Page Application) and mobile apps.
Ideal for	Ideal for modern web development and native-rendered apps for Android and iOS devices.	Ideal to use when you want to develop large-scale, feature-rich applications.
Model	It is based on Virtual DOM	Based on MVC (Model View Controller)
Written in	JavaScript	Typescript
Community Support	Facebook developers community	A large community of developers and supporters
Language preference	JSX - JavaScript XML	TypeScript
Companies Using	Facebook, Uber Technologies, Instagram, Netflix, Pinterest, etc.	Wepay, Beam, Auto Trader, Mesh, Streamline Social, etc.
Template	JSX + J% (ES5/ES6)	HTML + TypeScript
Abstraction	Strong	Medium

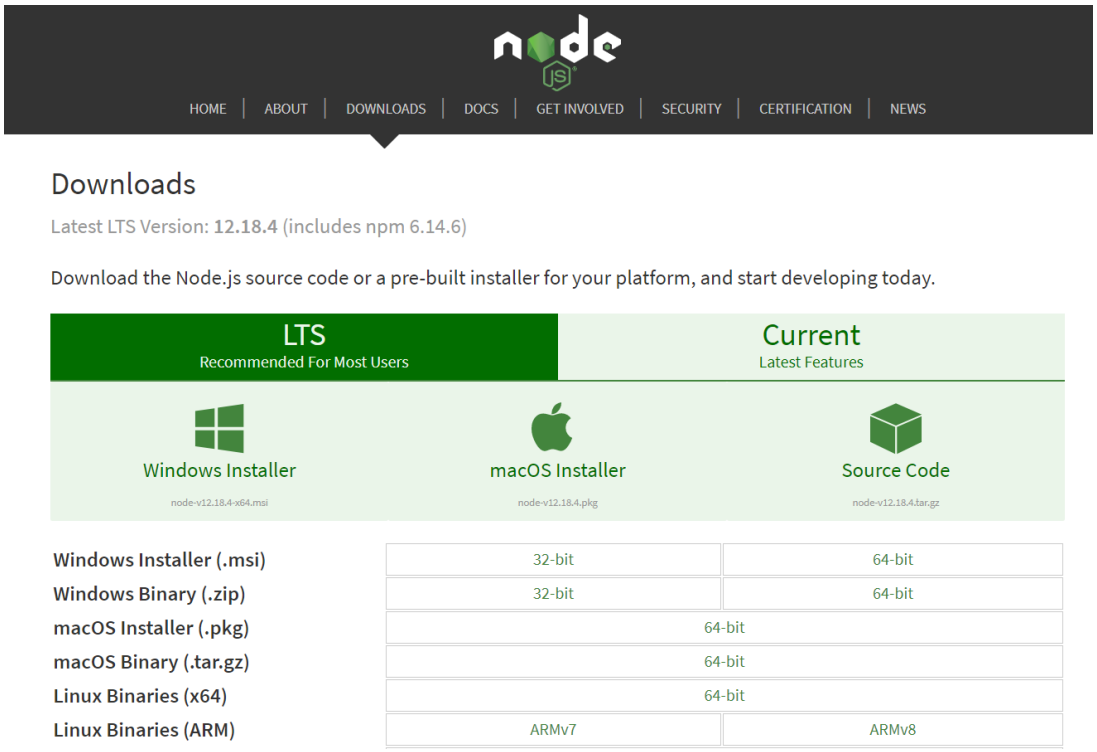
Git hub stars	126K	46.6 K
Adding Javascript library to the source code	Possible	Not possible
Restriction	React gives you an option to choose without putting any performance penalty.	An angular framework is very sensitive, which means that it restricts you from using large models.
Use of code	React allows you to manage the code according to your desired format.	Angular comes with many ready to use elements. However, it mainly comes from a specific provider. So, there are priority collisions and namesDaces.
Git hub stars	180K	80.8 K
Fork	30.3 K	48.2 K

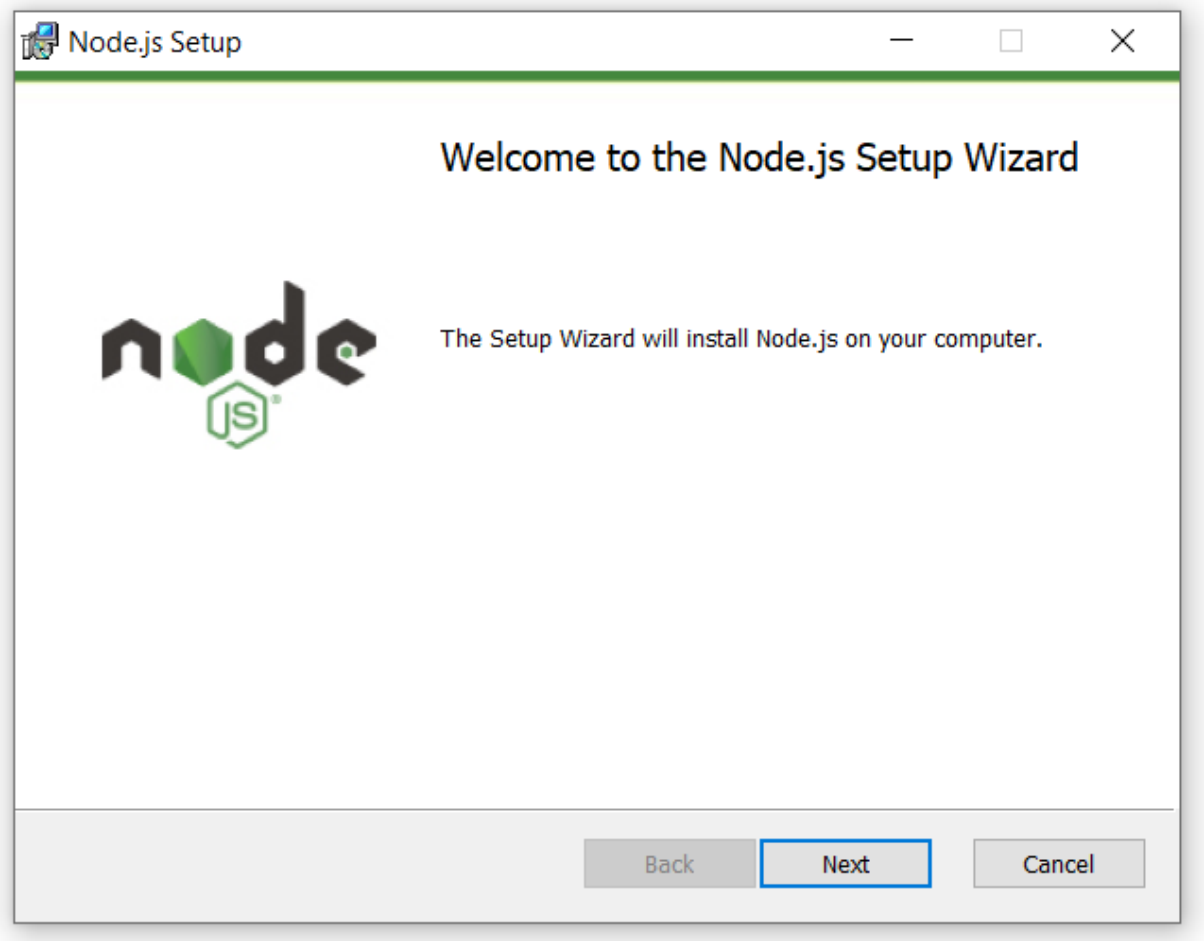
Which is Better?

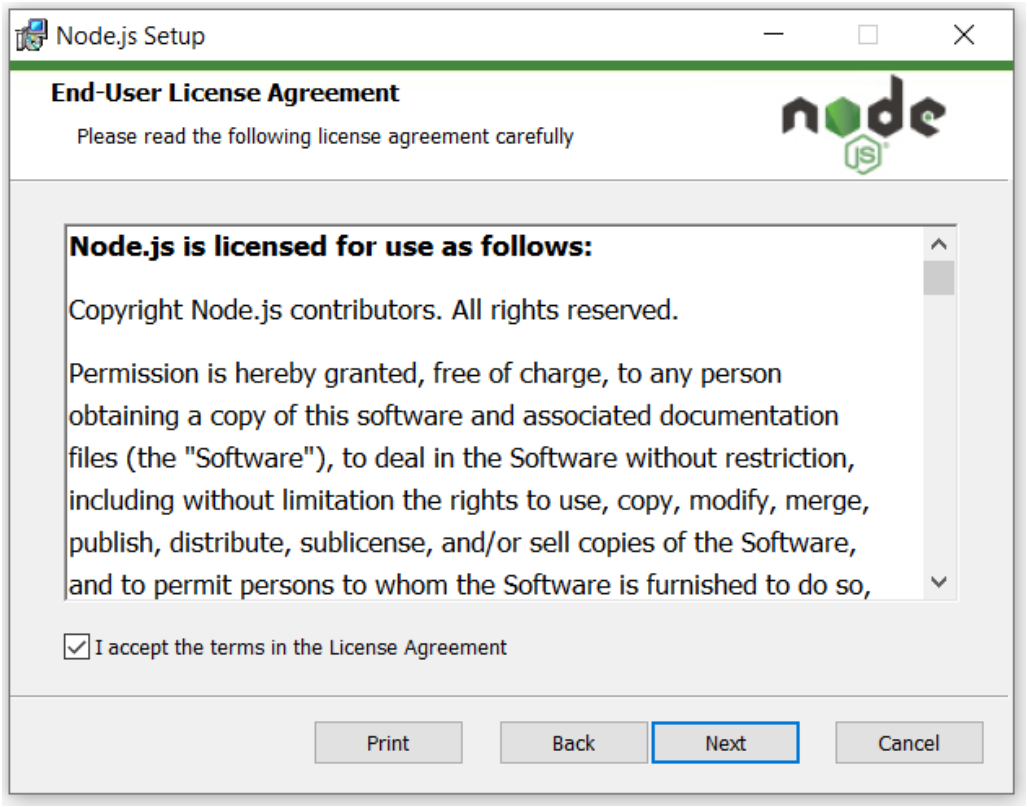
Both React and AngularJS are great options with respect to single page applications. However, both of them are also entirely different instruments. There might be statements like React is better than Angular or also vice versa.

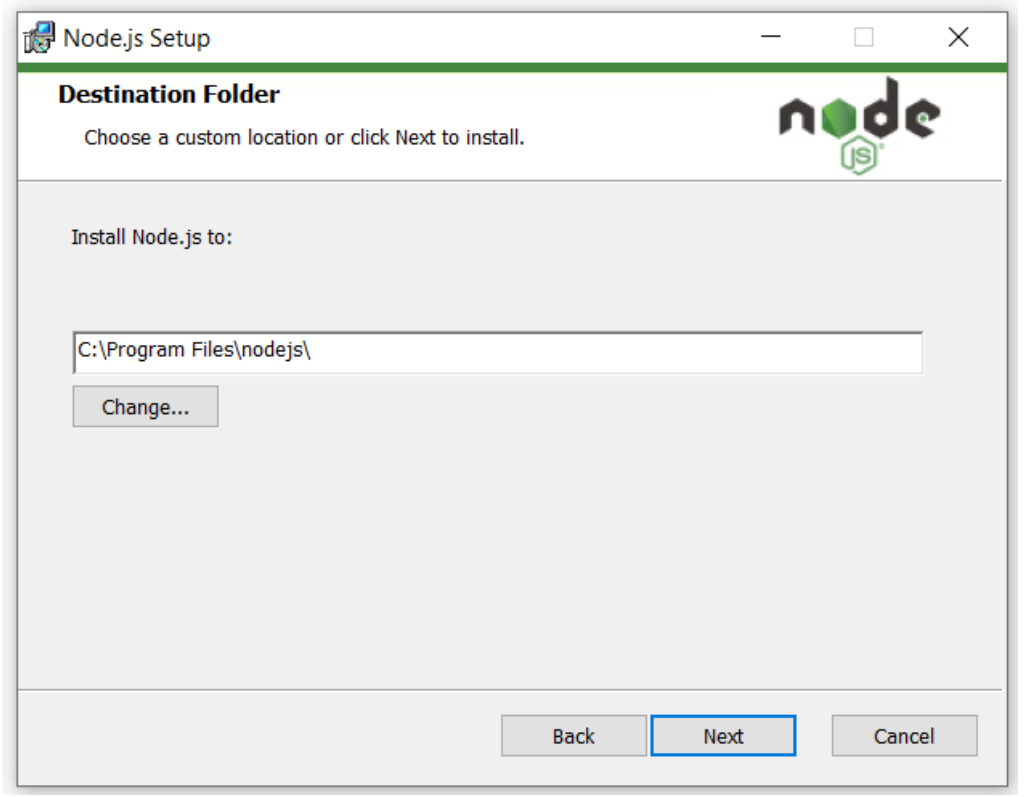
Whatever may be your perception of the discussion about React Vs. AngularJS, you need to make choices based on your, requirement of functionality and usability.

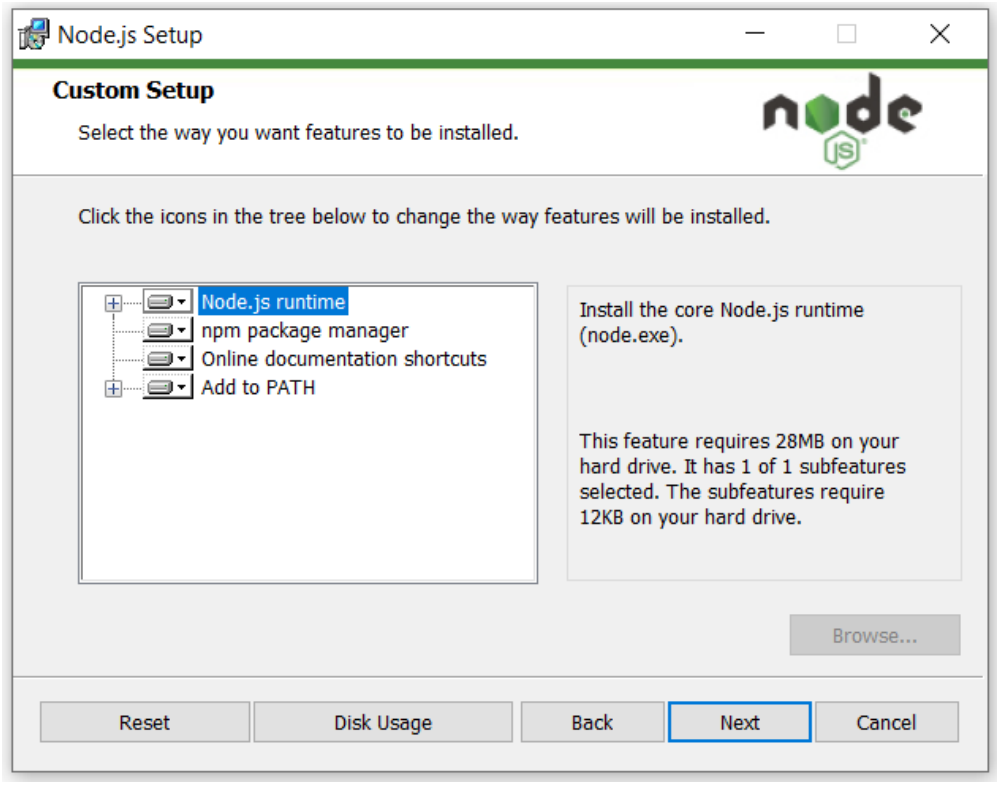
NodeJS Installation:

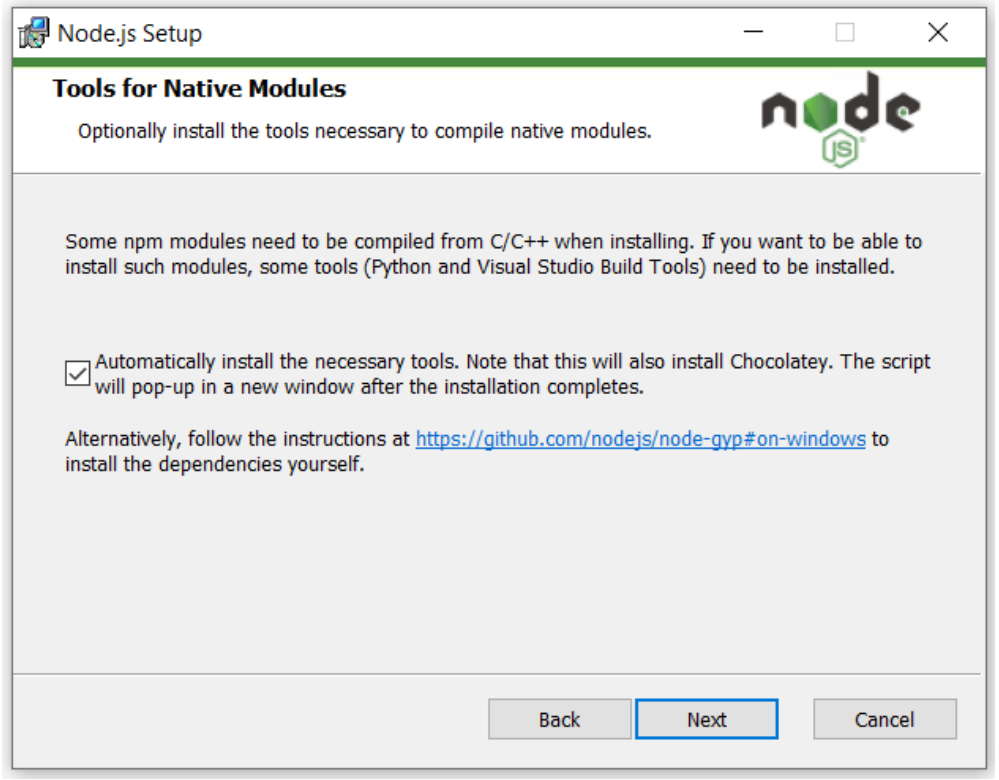


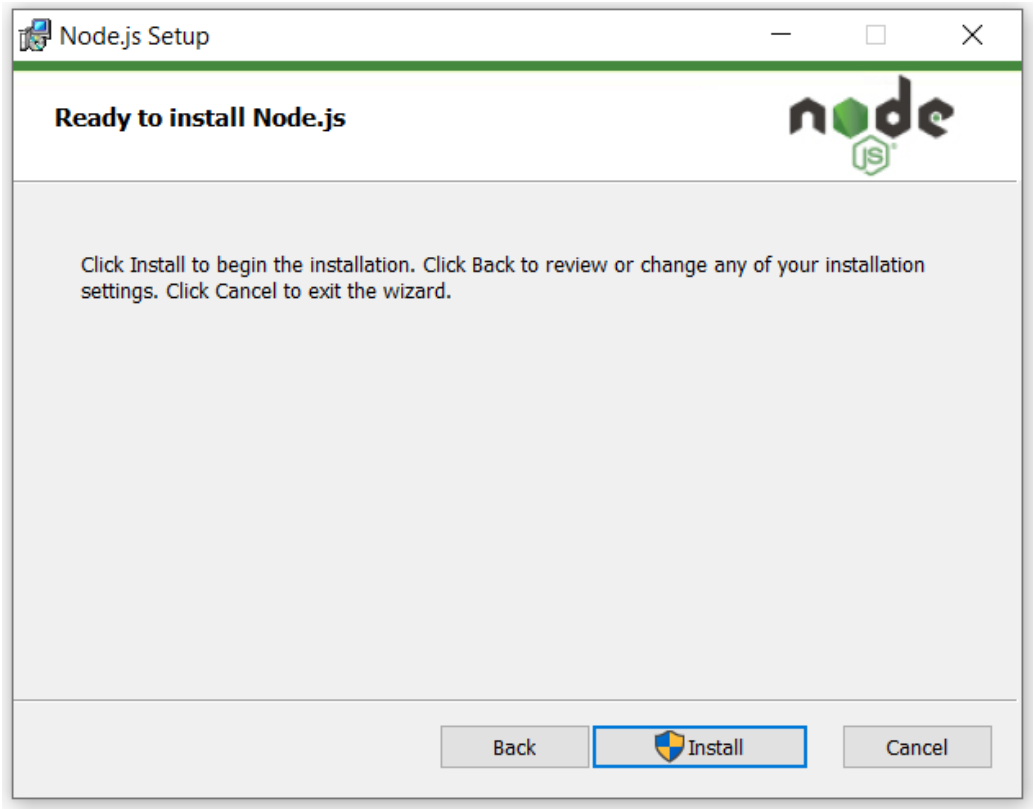


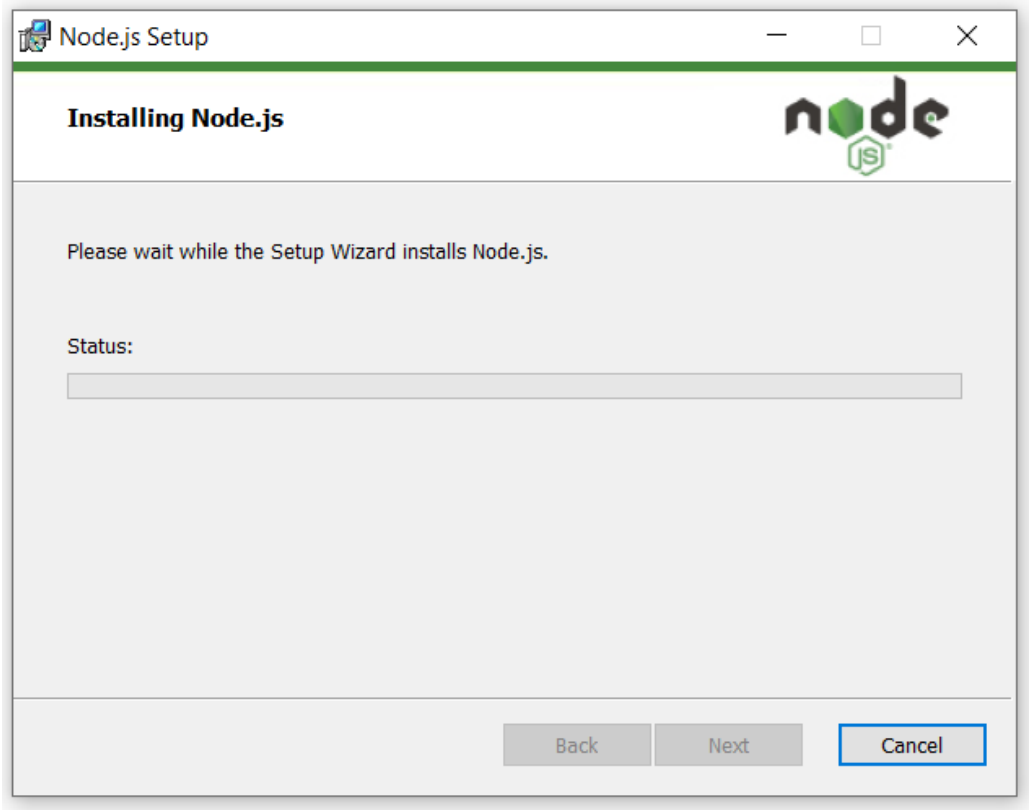

Node.js Setup
End-User License Agreement
Please read the following license agreement carefully
Node.js is licensed for use as follows:
Copyright Node.js contributors. All rights reserved.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so,
I accept the terms in the License Agreement
Print
Back
Next
Cancel

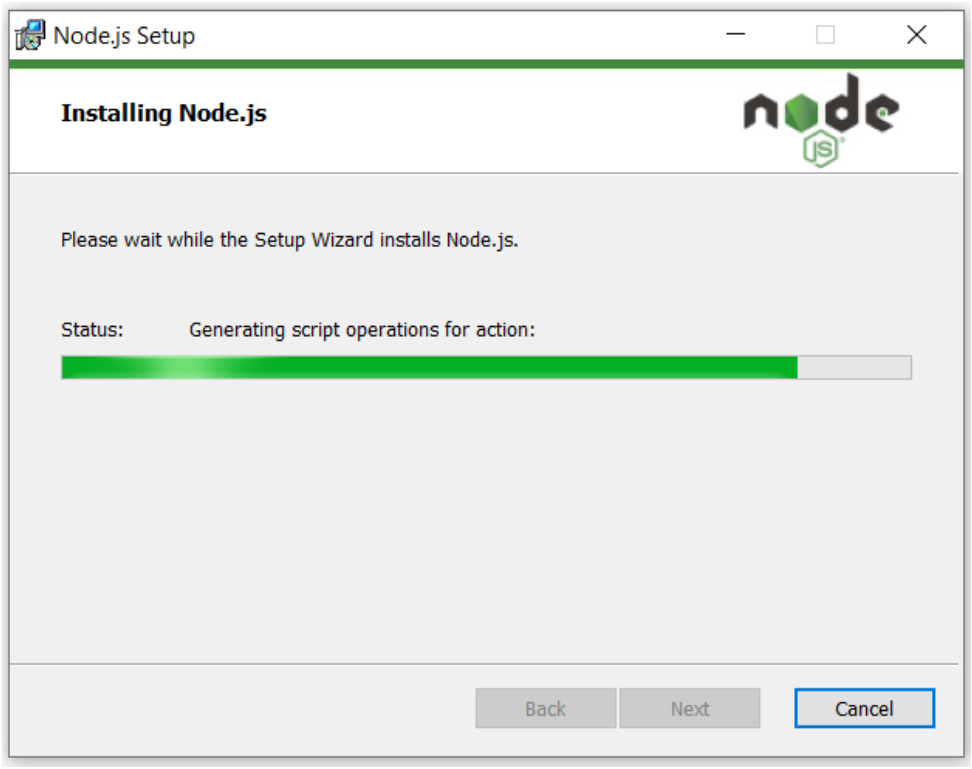

Node.js Setup
Destination Folder
Choose a custom location or click Next to install.
Install Node.js to:
C:\Program Files\nodejs\
Change...
Back
Next
Cancel

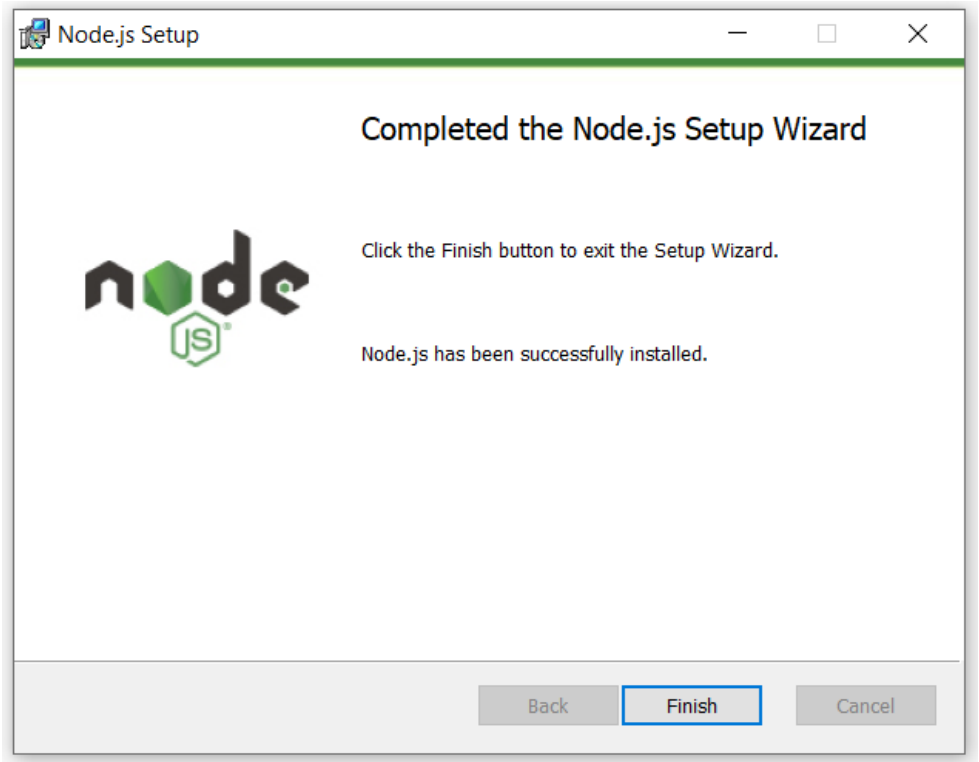

Node.js Setup
Custom Setup
Select the way you want features to be installed.
Click the icons in the tree below to change the way features will be installed.
Node.js runtime
npm package manager
Online documentation shortcuts
Add to PATH
Install the core Node.js runtime (node.exe).
This feature requires 28MB on your hard drive. It has 1 of 1 subfeatures selected. The subfeatures require 12KB on your hard drive.
Browse...
Reset
Disk Usage
Back
Next
Cancel


Node.js Setup
Tools for Native Modules
Optionally install the tools necessary to compile native modules.
Some npm modules need to be compiled from C/C++ when installing. If you want to be able to install such modules, some tools (Python and Visual Studio Build Tools) need to be installed.
Automatically install the necessary tools. Note that this will also install Chocolatey. The script will pop-up in a new window after the installation completes.
Alternatively, follow the instructions at https://github.com/nodejs/node-gyp#on-windows to install the dependencies yourself.
Back
Next
Cancel


Node.js Setup
Ready to install Node.js
Click Install to begin the installation. Click Back to review or change any of your installation settings. Click Cancel to exit the wizard.
Back
Install
Cancel


Node.js Setup
Installing Node.js
Please wait while the Setup Wizard installs Node.js.
Status:
Back
Next
Cancel


Node.js Setup
Installing Node.js
Please wait while the Setup Wizard installs Node.js.
Status: Generating script operations for action:
Back
Next
Cancel


Node.js Setup
Completed the Node.js Setup Wizard
Click the Finish button to exit the Setup Wizard.
Node.js has been successfully installed.
Back
Finish
Cancel

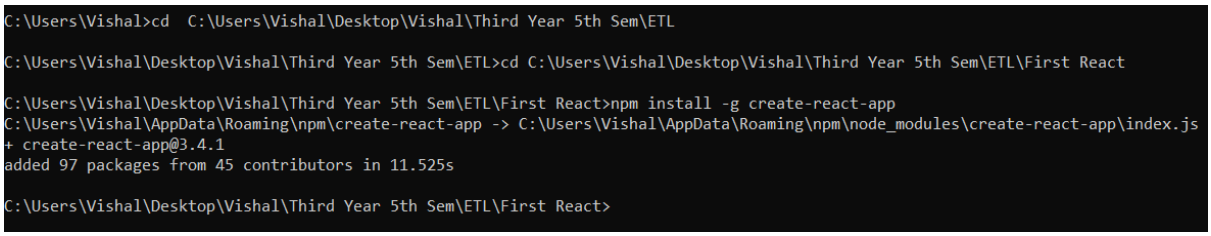
React Installation:

Step 1: To get started, we need to open a command prompt using Win+R and type '*cmd*'. Then, type in the following command.

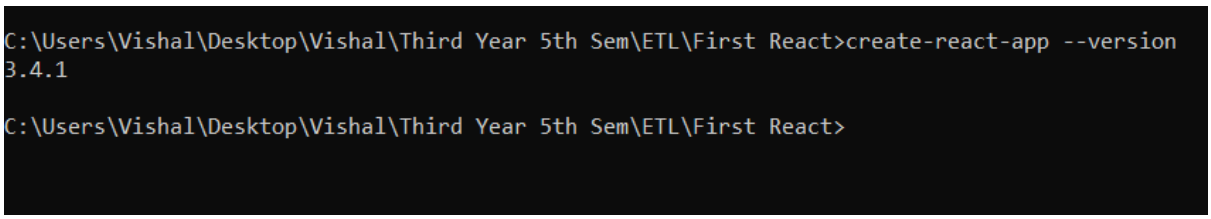
```
npm install -g create-react-app
```

This installs the *Create-React-App* module which makes it effortless to create and deploy React into projects with a single command.

```
npm install -g create-react-app
```



To check version use create-react-app --version



To create a new project and deploy React into it, we will run the following command to create **reactproject2**.

```
create-react-app reactproject2
```

The entire process is automated and begins with creating a new React app folder for the project, then installs packages and dependencies. The default packages include **react**, **react-dom**, and **react-scripts**. The installation will take a few minutes.

```
C:\Users\Vishal\Desktop\Vishal\Third Year 5th Sem\ETL\First React>create-react-app myapp

Creating a new React app in C:\Users\Vishal\Desktop\Vishal\Third Year 5th Sem\ETL\First React\myapp.

Installing packages. This might take a couple of minutes.
Installing react, react-dom, and react-scripts with cra-template...

[..................] | loadDep:signal-exit: sill resolveWithNewModule node-modules-regexp@1.0.0 checking installable status
```

```
C:\Users\Vishal\Desktop\Vishal\Third Year 5th Sem\ETL\First React>create-react-app myapp

Creating a new React app in C:\Users\Vishal\Desktop\Vishal\Third Year 5th Sem\ETL\First React\myapp.

Installing packages. This might take a couple of minutes.
Installing react, react-dom, and react-scripts with cra-template...


> core-js@2.6.11 postinstall C:\Users\Vishal\Desktop\Vishal\Third Year 5th Sem\ETL\First React\myapp\node_modules\babel-runtime\node_modules\core-js
> node -e "try{require('./postinstall')}catch(e){}"


> core-js@3.6.5 postinstall C:\Users\Vishal\Desktop\Vishal\Third Year 5th Sem\ETL\First React\myapp\node_modules\core-js
> node -e "try{require('./postinstall')}catch(e){}"


> core-js-pure@3.6.5 postinstall C:\Users\Vishal\Desktop\Vishal\Third Year 5th Sem\ETL\First React\myapp\node_modules\core-js-pure
> node -e "try{require('./postinstall')}catch(e){}"

+ cra-template@1.0.3
+ react@16.13.1
+ react-scripts@3.4.3
+ react-dom@16.13.1
added 1596 packages from 745 contributors and audited 1600 packages in 544.401s

71 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities

Git repo not initialized Error: Command failed: git --version
    at checkExecSyncError (child_process.js:630:11)
    at execSync (child_process.js:666:15)
    at tryGitInit (C:\Users\Vishal\Desktop\Vishal\Third Year 5th Sem\ETL\First React\myapp\node_modules\react-scripts\scripts\init.js:46:5)
    at module.exports (C:\Users\Vishal\Desktop\Vishal\Third Year 5th Sem\ETL\First React\myapp\node_modules\react-scripts\scripts\init.js:266:7)
    at [eval]:3:14
    at Script.runInThisContext (vm.js:120:18)
    at Object.runInThisContext (vm.js:309:38)
    at Object.<anonymous> ([eval]-wrapper:10:26)
    at Module._compile (internal/modules/cjs/loader.js:1137:30)
    at evalScript (internal/process/execution.js:94:25) {
  status: 1,
  signal: null,
  output: [ null, null, null ],
  pid: 10740,
  stdout: null,
  stderr: null
}
```

```
Installing template dependencies using npm...
npm WARN tsutils@3.17.1 requires a peer of typescript@>=2.8.0 || >= 3.2.0-dev || >= 3.3.0-dev || >= 3.4.0-dev || >= 3.5.0-dev || >= 3.6.0-dev || >= 3.6.0-beta || >= 3.7.0-dev || >= 3.7.0-beta but none is install
ed. You must install peer dependencies yourself.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@2.1.2 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.1.2: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.13 (node_modules\jest-haste-map\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.13: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.13 (node_modules\watchpack-chokidar2\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.13: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.13 (node_modules\webpack-dev-server\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.13: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})

+ @testing-library/jest-dom@4.2.4
+ @testing-library/react@9.5.0
+ @testing-library/user-event@7.2.1
added 36 packages from 56 contributors and audited 1636 packages in 119.376s

71 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities

Removing template package using npm...

npm WARN tsutils@3.17.1 requires a peer of typescript@>=2.8.0 || >= 3.2.0-dev || >= 3.3.0-dev || >= 3.4.0-dev || >= 3.5.0-dev || >= 3.6.0-dev || >= 3.6.0-beta || >= 3.7.0-dev || >= 3.7.0-beta but none is install
ed. You must install peer dependencies yourself.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@2.1.2 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.1.2: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.13 (node_modules\watchpack-chokidar2\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.13: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.13 (node_modules\jest-haste-map\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.13: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.13 (node_modules\webpack-dev-server\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.13: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})

removed 1 package and audited 1635 packages in 22.184s

71 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities


Success! Created myapp at C:\Users\Vishal\Desktop\Vishal\Third Year 5th Sem\ETL\First React\myapp
Inside that directory, you can run several commands:

  npm start
    Starts the development server.
```

```
71 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities


Success! Created myapp at C:\Users\Vishal\Desktop\Vishal\Third Year 5th Sem\ETL\First React\myapp
Inside that directory, you can run several commands:

  npm start
    Starts the development server.

  npm run build
    Bundles the app into static files for production.

  npm test
    Starts the test runner.

  npm run eject
    Removes this tool and copies build dependencies, configuration files
    and scripts into the app directory. If you do this, you can’t go back!

We suggest that you begin by typing:

  cd myapp
  npm start

Happy hacking!

C:\Users\Vishal\Desktop\Vishal\Third Year 5th Sem\ETL\First React>
```

```
C:\Users\Vishal\Desktop\Vishal\Third Year 5th Sem\ETL\First React>cd myapp

C:\Users\Vishal\Desktop\Vishal\Third Year 5th Sem\ETL\First React\myapp>npm start

> myapp@0.1.0 start C:\Users\Vishal\Desktop\Vishal\Third Year 5th Sem\ETL\First React\myapp
> react-scripts start

i ｢wds｣: Project is running at http://192.168.56.1/
i ｢wds｣: webpack output is served from
i ｢wds｣: Content not from webpack is served from C:\Users\Vishal\Desktop\Vishal\Third Year 5th Sem\ETL\First React\myapp\public
i ｢wds｣: 404s will fallback to /
Starting the development server...
Compiled successfully!

You can now view myapp in the browser.

  Local:            http://localhost:3000
  On Your Network:  http://192.168.56.1:3000

Note that the development build is not optimized.
To create a production build, use npm run build.
```

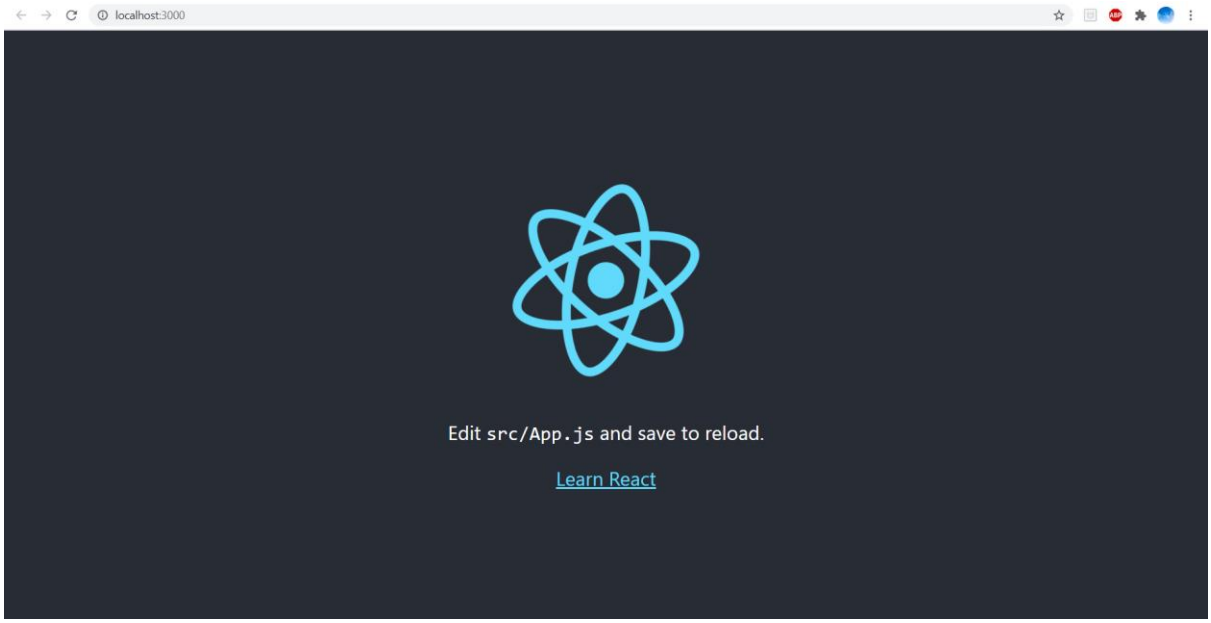
```
Compiled successfully!

You can now view myapp in the browser.

  Local:            http://localhost:3000
  On Your Network:  http://192.168.56.1:3000

Note that the development build is not optimized.
To create a production build, use npm run build.
```

Web Browser:



Problem Definition:

This application is developed for educational purpose, allowing the users to prepare the multiple-choice questions for different examinations. The main goal of the application is to enable users to practice for MCQ tests. Our application display score of user how many questions are right from test at the end of the result and also display at the time of submission of answer that selected answer is correct or not. After end of the test if user want to mail the score which is secured in test then user can click on Send mail button and user get email that how much score user get.

Also, login and registration are there to keep track of user to know which user score how many marks in respective quiz.

Conclusion:

Thus, From the above experiment we understand use of Nodejs, Angularjs, Reactjs etc. as well as how to install them. Learn advantages and disadvantage of framework and compare them.