This document having complete resources needed for the **Full Stack Web Development and System Design Interviews.**

This document is made up of multiple hyperlinks – to access them – **right click > open hyperlink.**

As human it is common for all of us to make mistakes, even I’ve made some while writing this doc – If there are any such mistakes in the document – please kindly , send us a email.

* Use the guided Productivity hacks mentioned in the last Page.
* Updated versions of the document will be available in the git- repository mentioned in the last page.

**Web Development Learning Resources: From Beginner to Advanced.**

[**MindMap – Complete**](https://coggle.it/diagram/YJ20erGfmnpJMhLQ/t/full-stack-web-development-beginner-to-interview-ready)

[Web development](https://en.wikipedia.org/wiki/Web_development) refers to the act of building and maintaining websites; these websites can range in complexity from simple static websites to more complex web applications with huge databases, real-time data-intensive applications such as social media and eCommerce applications.

**Back-end developers** designs the application APIs and databases schemas and implements the business logic. Backend developers were originally referred to as developers back in the days when front-end development was considered to be trivial work. Backend developers can employ a wide range of languages to achieve their needs. A backend developer would be skilled in languages such as PHP, Java, Python, ASP.net, Ruby, etc.

**Front-end developers** consume the APIs and implement the logic to handle a website’s presentation, interactivity, and user experience. Unlike backend, the front-end developer’s main scripting language is JavaScript. However, there is an overwhelming amount of JavaScript frameworks, libraries, bundlers, and build tools available for front-end development. And oftentimes a front-end developer is specialised in one or more of these. Some examples of these front-end technologies are **React**, Angular, Vue, Svelte, Webpack, Rollup, Snowpack, ESbuild, etc.

Besides JavaScript, front-end developers also use HTML and CSS to both structure and beautify a website.

**Full-stack developers** are developers who has both front-end and back-end development skills.

Regardless of the area of specialisation, good developers follow the [boy/girl scout rule](https://www.stepsize.com/blog/how-to-be-an-effective-boy-girl-scout-engineer?utm_source=dev.to&utm_medium=referral): always leave the code better than you found it. And they achieve this by using [tools that enhance productivity](https://dev.to/alexomeyer/10-must-have-vs-code-extensions-to-improve-your-productivity-4goe), improve code quality, foster collaboration, and manage technical debt.

I’ll be mentioning few blogs in different places, - that will help you in creating your own way to start ( explore all this roadmap blogs so that you’ll understand things completely. )

1. [ROAD MAP](https://roadmap.sh/) :

**Roadmap.sh** is a community effort to create roadmaps, guides, and other educational content to help guide the developers in picking up the path and guide their learnings.

Read map provides downloadable visual guides containing topics, tools, languages, and all the required niceties to guide a noob developer in his journey to web development mastery.

One has to be clear with INTERNET Fundamentals to start, this will help along the way – ( Don’t try to mug up while watching these long lectures – this computer networking concepts will you glimpse of how to internet works in general.. Hey, curious right? – Start them and notes if there is anything important about the protocols or anything that surprised you.. )

* [How Does the Internet Work?](https://www.youtube.com/watch?v=TNQsmPf24go)  ( MVP )
* [Computer Network Fundamentals By Kunal Kushwaha.](https://www.youtube.com/watch?v=IPvYjXCsTg8)

Do you want more? , Curious to learn it completely?? – Let’s Start!!

[**CS144, Introduction to Computer Networking**](https://www.youtube.com/playlist?list=PLoCMsyE1cvdWKsLVyf6cPwCLDIZnOj0NS) **( This playlist MVP )**

**So, hoping that you are having good understanding of Internet ( Computer Networking. )**

**Front- End Development :**

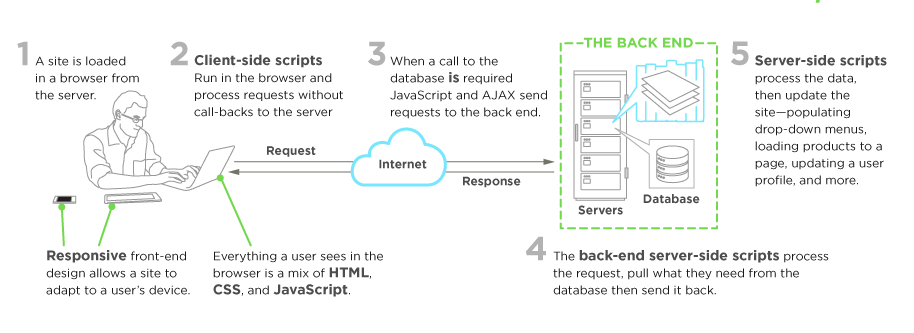
**Follow the path even if you are a complete beginner.**

Front-end web development, also known as **client-side development is the practice of producing HTML, CSS and JavaScript for a website or Web Application** so that a user can see and interact with them directly. The challenge associated with front end development is that the tools and techniques used to create the front end of a website change constantly and so the developer needs to constantly be aware of how the field is developing.

The objective of designing a site is to ensure that when the users open up the site they see the information in a format that is easy to read and relevant. This is further complicated by the fact that users now use a large variety of devices with varying screen sizes and resolutions thus forcing the designer to take into consideration these aspects when designing the site. They need to ensure that their site comes up correctly in different browsers (cross-browser), different operating systems (cross-platform) and different devices **(cross-device), which requires careful planning on the side of the developer.**

**HTML, CSS, & JavaScript:**

A front-end developer architects and develops websites and applications using web technologies (i.e., [HTML](https://developer.mozilla.org/en-US/docs/Web/HTML), [CSS](https://developer.mozilla.org/en-US/docs/Web/CSS), [DOM](https://developer.mozilla.org/en-US/docs/Web/API/Document_Object_Model), and [JavaScript](https://developer.mozilla.org/en-US/docs/Web/JavaScript)), which run on the [Open Web Platform](https://en.wikipedia.org/wiki/Open_Web_Platform) or act as compilation input for non-web platform environments (i.e., [React Native](https://facebook.github.io/react-native/)).



**HTML ( Hypertext Markup Language ):**

Here are some resources to begin with.. ( Learn while doing – Don’t watch videos without installing vs code ( any code-editor of your choice )’

What is a code editor?? – Google!!!

[**Visual Studio Code Tutorial for Beginners - Introduction**](https://www.youtube.com/watch?v=VqCgcpAypFQ)– So, jump start creating things!!!

* [**Brad’s Boot Camp**](https://www.youtube.com/watch?v=HqzWCNbX_wg&list=PLpcSpRrAaOargYaCNYxZCiFIp9YTqEl-l) \_ ( No, you don’t have to complete it, just watch first 5 videos in playlist– video 5 by the end of 5th video you should be able to post your first static web page – using GitHub pages, so be cautious – don’t just go through videos – start doing!!! – in codepen.io.. )

Yea, I know you got this doubt right? , “ What is this GitHub??’

Finish creating your first static page ( till 5th video in the mentioned bootcamp ) will learn github too..

* Yeaaaaaa…., you got that right! One has to have a basic understanding of GitHub if you are in the field..

This is for you..

[**Git/ Github Tutorial – In Depth.**](https://www.youtube.com/watch?v=xAAmje1H9YM&list=PLeo1K3hjS3usJuxZZUBdjAcilgfQHkRzW) **( Try to execute the commands while you are going through videos ) – MVP**

Not so clear?? , Go through this [**Complete Git and GitHub Tutorial**](https://d.docs.live.net/671f5e0f27ed51d2/Desktop/Documents/Complete%20Git%20and%20GitHub%20Tutorial)

* [**HTML FULL COURSE.**](https://www.youtube.com/watch?v=7mdEvKyLoYM)( While going through video – I suggest you to create the html files and upload them to a repository and write the documentation for the files you’ve uploaded using Read me file. )

**Blogs – HTML :**

* [**https://www.w3schools.com/html/**](https://www.w3schools.com/html/)
* [**https://www.freecodecamp.org/**](https://www.freecodecamp.org/)
* [**https://www.codecademy.com/learn/learn-html**](https://www.codecademy.com/learn/learn-html)
* [**https://www.oreilly.com/library/view/learning-web-design/9781449337513/ch04.html**](https://www.oreilly.com/library/view/learning-web-design/9781449337513/ch04.html)
* [**https://developer.mozilla.org/en-US/docs/Learn/HTML/Introduction\_to\_HTML/Document\_and\_website\_structure**](https://developer.mozilla.org/en-US/docs/Learn/HTML/Introduction_to_HTML/Document_and_website_structure)
* [**https://www.tutorialspoint.com/html/index.htm**](https://www.tutorialspoint.com/html/index.htm)

**Tutorials :** ( Refer one among the mentioned tutorials , if you aren’t clear )

* [**Freecodecamp**](https://www.youtube.com/watch?v=pQN-pnXPaVg)
* [**Traversy Media**](https://www.youtube.com/watch?v=UB1O30fR-EE)
* [**KodeGod**](https://www.youtube.com/watch?v=yhASPkhITgQ)
* [**Udemy Youtube**](https://www.youtube.com/watch?v=BU2rjqA_8Y4&list=PL6cactdCCnTLkQah9GKzsJmiLbegy4dEk)
* [**Scrimba**](https://scrimba.com/learn/htmlcss)

**CSS ( Cascading Style Sheets ) :**

**Cascading Style Sheets**, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable. CSS handles the look and feel part of a web page. Using CSS, you can control the colour of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colours are used, layout designs, variations in display for different devices and screen sizes as well as a variety of other effects.

CSS is easy to learn and understand but it provides powerful control over the presentation of an HTML document. Most commonly, CSS is combined with the markup languages HTML or XHTML.

Advantages of CSS :

* CSS saves time
* Pages load faster
* Easy maintenance
* Superior styles to HTML
* Multiple Device Compatibility
* Global web standards.

[**Frontend BootCamp**](https://www.youtube.com/watch?v=7TMescjxlrw&list=PLqcJACtjWm_VEtYJVIKDDmVWlxY-IWGDj) **:** Yea, I know they are too long right? – Still, prefer doing this playlist because it helps you in getting in - depth understanding in html, css if you complete till Day -9 in the playlist!

But, be as much as curious as possible.. – Keep doing the work pratically..

**Approach:** Start the video – Do the same implementation in your code editors while watching videos – Upload one by one files with complete documentation ( I mean write what you’ve done in the file ) in read me readme.md file.

* [**CSS complete playlist by Apna College ( Hindi )**](https://www.youtube.com/watch?v=sqJ6xZ9mUwE&list=PL4PaOIHsZcwJ5WLSMeOCrlmrLdphfhcpo)
* [**Html-CSS by codedamn**](https://www.youtube.com/watch?v=XhqEuyWjbdo)
* [**Whole Advanced CSS Playlist – codegyaan ( Hindi )**](https://www.youtube.com/watch?v=HhMRVdgw67M&list=PL7akPJI4biSIqVTT7Fhk-6As5OgFVIKqD)

**Blogs – CSS :**

* <https://www.w3schools.com/css/>
* <https://www.tutorialspoint.com/css/what_is_css.htm>
* <https://scrimba.com/learn/htmlcss>
* <https://css-tricks.com/>
* <https://www.codecademy.com/learn/learn-css>
* <https://webdesign.tutsplus.com/tutorials/the-best-way-to-learn-css--webdesign-11906>

**Upskill with Advanced – CSS :**

Try solving things from **CSS BATTLE**.

Hey, what is this?? – Is css battle really helpful.. – Yeah, it’ll help you with all advanced concepts of css – makes you css pro.. – Though, there are n – number of libraries – but still has it’s place in the creative scenarios..

So, How to solve css battle challenges..

These are some example videos on the same..

* [Frontend Architect attempts the CSS Battle Challenge](https://www.youtube.com/watch?v=28-7YSC0_zw)
* [Can I Beat These Hard CSS Battles?](https://www.youtube.com/watch?v=9k1RcGb42t0) – **Conner Ardman.**

**Other Youtube videos** for the CSS : ( The above mentioned were more than enough for this )

* [Freecodecamp](https://www.youtube.com/watch?v=1Rs2ND1ryYc)
* [Traversy Media](https://www.youtube.com/watch?v=yfoY53QXEnI)
* [Academind](https://www.youtube.com/watch?v=wNX7lWzchow)
* [Derek Banas](https://www.youtube.com/watch?v=CUxH_rWSI1k)
* [Edureka](https://www.youtube.com/watch?v=3_9znKVNe5g)

**CSS Frameworks :**

* [Bootstrap 5 Tutorial For Beginners](https://www.youtube.com/watch?v=bzZyCB5QT9I&list=PLp50dWW_m40WzAh2wY0a9iFYb0QlPzoxx)

Learn from the official documentation. [Bootstrap · The most popular HTML, CSS, and JS library in the world. (getbootstrap.com)](https://getbootstrap.com/)

* [Bootstrap video by Apna College](https://www.youtube.com/watch?v=l1EssrLxt7E&list=PLfqMhTWNBTe3H6c9OGXb5_6wcc1Mca52n) ( Videos 28 to 31 )

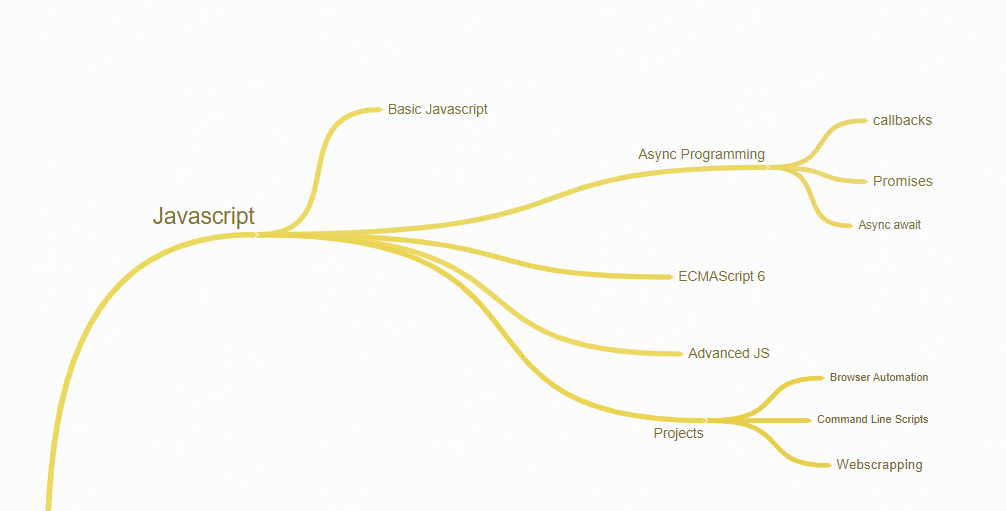
Other frameworks :

* [Tailwind CSS](https://tailwindcss.com/)
* [Semantic - UI](https://semantic-ui.com/)
* there are plenty of resources related to CSS frameworks – Refer documentation for more information related to tailwindcss.

**Vanilla JavaScript :**

JavaScript was initially created to “**make web pages alive**”. The programs in this language are called scripts. They can be written right in a web page’s HTML and run automatically as the page loads.

Scripts are provided and executed as plain text. They don’t need special preparation or compilation to run. In this aspect, JavaScript is very different from another language called Java.



[Confused where to start first??](https://www.youtube.com/watch?v=qyPKOB9bPG8&list=PL-Jc9J83PIiFU_evuYnTGHVXU_LeZryNP) ( This series of videos will help you with the basic understanding of JavaScript – Fundamentals! )

JavaScript tutorials :

* [**Practical approach to JavaScript**](https://www.youtube.com/watch?v=mrEcfu-ByDw&list=PL-Jc9J83PIiGT2wmeqRM6ZcjdsqlFvDMz) ( This is the only playlist which had unique way to teach stuff in js – go through – try to implement the same while watching these videos )
* **Had a cleared understanding of promises, Async- Await, High order functions, call back functions - right??**
* If you are not confident with the above concepts – I suggest you watch Namaste JavaScript playlists – these two playlists help you with all the core-javascript fundamentals..

Caution: **You’ll fell in JavaScript!!!**

* [Namaste JavaScript Season -1](https://www.youtube.com/watch?v=pN6jk0uUrD8&list=PLlasXeu85E9cQ32gLCvAvr9vNaUccPVNP)
* [Namaste JavaScript Season -2](https://www.youtube.com/watch?v=78Homn79Qsk&list=PLlasXeu85E9eWOpw9jxHOQyGMRiBZ60aX)

**Courses on JavaScript** :[**A comprehensive approach to implement JavaScript concepts. – Try to implement everything discussed in these videos ( Upload them to your git repo ) – VEDIOS 10 -22**](https://www.youtube.com/watch?v=7TMescjxlrw&list=PLqcJACtjWm_VEtYJVIKDDmVWlxY-IWGDj)

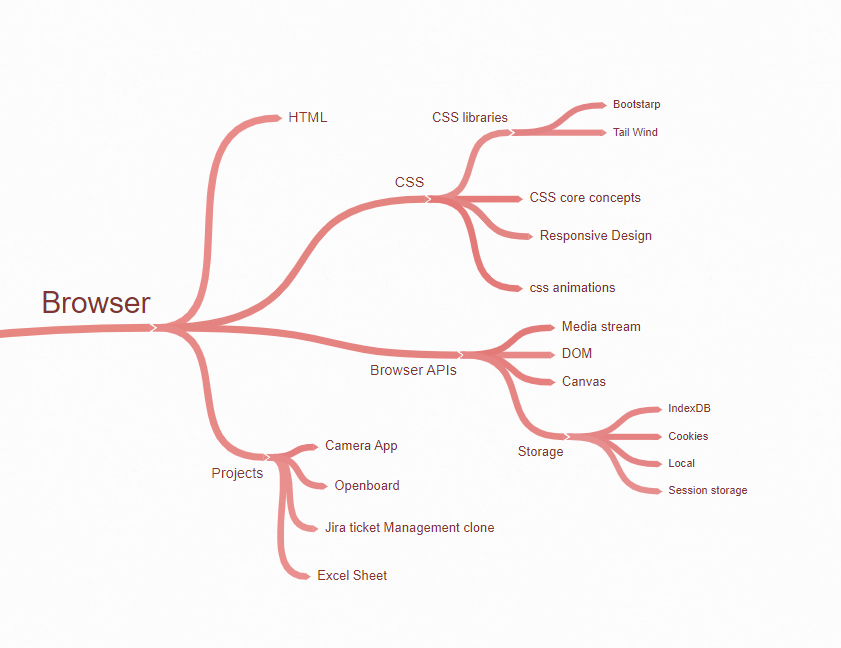
Other video tutorials :

* [Edureka](https://www.youtube.com/watch?v=o1IaduQICO0)
* [Clever Programmer](https://www.youtube.com/watch?v=Qqx_wzMmFeA)
* [CodeSTACkr](https://www.youtube.com/watch?v=d5ob3WAGeZE)
* [Freecodecamp](https://www.youtube.com/watch?v=PkZNo7MFNFg)
* [Traversy Media](https://www.youtube.com/watch?v=hdI2bqOjy3c)

Blog – Documentations :

* <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
* <https://devdocs.io/javascript/>
* <https://www.w3schools.com/js/>
* <https://javascript.info/document>
* <https://documentation.js.org/>

Clear with browser fundamentals? – Pepcoding youtube channel..



**JavaScript Frameworks :**

As a programmer, you don’t need to start from scratch when you have tools designed to help you with your projects. Frameworks are software that is developed and used by developers to build applications.

**What is Frameworks?**

Since they are often built, tested, and optimized by several experienced software engineers and programmers, software frameworks are versatile, robust, and efficient. Using a software framework to develop applications lets you focus on the high-level functionality of the application. This is because any low-level functionality is taken care of by the framework itself.

**Why do we use Frameworks?**

Developing software is a complex process. It necessitates a plethora of tasks, including coding, designing, and testing. For only the coding part, programmers had to take care of the syntax, declarations, garbage collection, statements, exceptions, and more.

Software frameworks make life easier for developers by allowing them to take control of the entire software development process, or most of it, from a single platform.

**Advantages of using a software framework:**

* Assists in establishing better programming practices and fitting use of design patterns.
* Code is more secure.
* Duplicate and redundant code can be avoided .
* Helps consistent developing code with fewer bugs.
* Makes it easier to work on sophisticated technologies.
* One could create their software framework or contribute to open-source frameworks. Hence, there is a continuous improvement in the functionality.
* Several code segments and functionalities are pre-built and pre-tested. This makes applications more reliable.
* Testing and debugging the code is a lot easier and can be done even by developers who do not own the code.
* The time required to develop an application is reduced significantly.

**There are two type of frameworks.**

**1. Frontend**

**2. backend**

**Frontend frameworks :**

* [ReactJs](https://reactjs.org/)
* [Angular](https://angularjs.org/)
* [VueJs](https://vuejs.org/)
* [EmberJs](https://emberjs.com/)
* [Backbone Js](https://backbonejs.org/)

**BackendJs :**

* [NextJs](https://nextjs.org/)
* [Gatsby](https://www.gatsbyjs.com/)
* [NodeJs](https://nodejs.org/en/)
* [ExpressJs](https://expressjs.com/)
* [MeteorJs](https://www.meteor.com/)

**Framework resources -**

As I personally prefer to use react for the frontend and node or

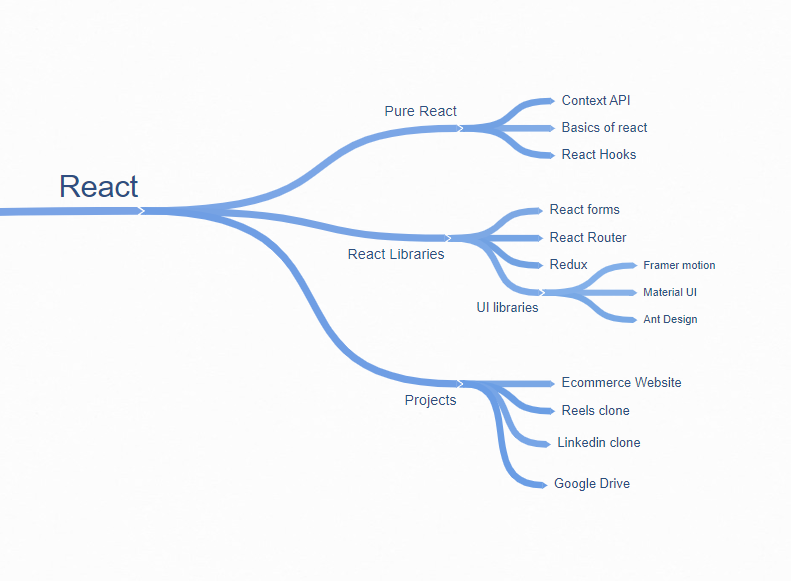
Next js for the backend. I think it is the best combo generally

prefer by the all the developers.

So, here I am provide some resource for REACT and

NEXT/NODE.

**ReactJs :**

****

Firstly, revise ES6 concepts thoroughly..

To get the gist of reactjs – watch this [**video by Mosh**](https://www.youtube.com/watch?v=Ke90Tje7VS0) first – But, I suggest you to directly jump directly to the next step.

Then learn while making projects :

* [**This Playlist ( vedios 24 - 36 ) will help you to build Commerce App and Netflix,Spotify Clone**](https://www.youtube.com/watch?v=7TMescjxlrw&list=PLqcJACtjWm_VEtYJVIKDDmVWlxY-IWGDj&index=1)

**Now, BUILD BUILD BUILD –** try to create as many websites as possible the knowledge you’ve gained.

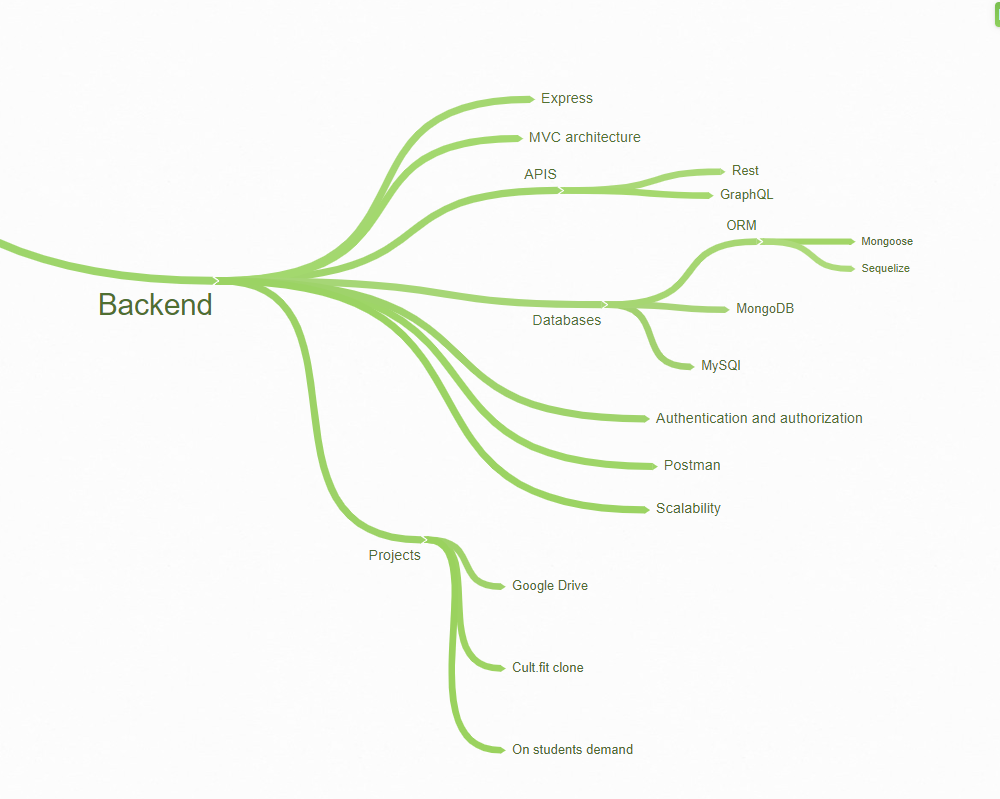
**Youtube Channels and Playlists of React as a Whole :**

* **Freecodecamp** - [Learn Fundamentals, Hooks, Context API, React Router, Custom Hooks.](https://www.youtube.com/watch?app=desktop&v=4UZrsTqkcW4&feature=youtu.be)
* [**Intermediate React and Firebase Tutorial - Build an Evernote Clone**](https://www.youtube.com/watch?app=desktop&v=I250xdtUvy8&feature=youtu.be)
* [Basic React things](https://www.youtube.com/watch?v=gY5sGvq-8h8)
* [Basics including useState, Components, Props](https://www.youtube.com/watch?v=CKH5zLGBJAI&list=PLKhlp2qtUcSYOMyooUr3IIDF5TMN488l7).
* [Movies and TV Series Searching App in React JS and Material UI](https://www.youtube.com/watch?v=vxUfx4aM5d8&list=PLKhlp2qtUcSa_rX7glmB7HyFsEOEQa0Uk) - Beginner level project ( React )’
* [React Hooks](https://www.youtube.com/watch?v=4Cf86qVEIJY&list=PLKhlp2qtUcSYppHmY6mZ4WspqM-uPSk7T) – RoadSideCoder.
* [React Interview Preparation](https://www.youtube.com/watch?v=20F_KzHPpvI&list=PLKhlp2qtUcSZiWKJTi5-5r6IRdHhxP9ZU)
* [React JS - React Tutorial for Beginners in Hindi - TechGun](https://www.youtube.com/watch?v=piPwsPXY_a8&list=PLjpp5kBQLNTTz1BZXJfU6ZHWKeyzTkgCo)
* [ReactJS Tutorial for Beginners - Thapa Technical](https://www.youtube.com/watch?v=tiLWCNFzThE&list=PLwGdqUZWnOp3aROg4wypcRhZqJG3ajZWJ)
* [15 react Projects.](https://www.youtube.com/watch?app=desktop&v=a_7Z7C_JCyo&feature=youtu.be)

**Blogs – Docs :**

* <https://egghead.io/courses/the-beginner-s-guide-to-react>
* <https://ui.dev/free-react-bootcamp/>
* <https://www.newline.co/fullstack-react/30-days-of-react/>
* <https://www.freecodecamp.org/news/building-chrome-extensions-in-react-parcel-79d0240dd58f/>
* <https://scrimba.com/learn/learnreact>

**Backend Development :**



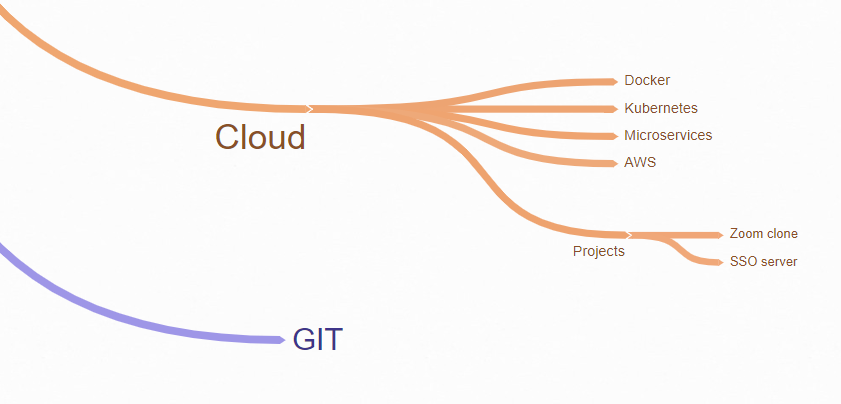
**NodeJS :**

* [NodeJs – Crash Course](https://www.youtube.com/watch?v=vJEO57B05Sg)
* [Complete Backend Development Course.](https://www.youtube.com/watch?v=fupTFbQv3MQ&list=PL-Jc9J83PIiEnK1q9tuVrrORqKBexcE_J)
* [Node.js Full Course for Beginners | Complete All-in-One Tutorial | 7 Hours](https://www.youtube.com/watch?v=f2EqECiTBL8)

These two videos are more than enough to understand the backend.

**But but but, please explore as many as resources possible in the process.**

Express Js, Mongo DB , Cloud ( Firebase, AWS ) and many more – do explore..



Most of them are already covered in the resources provided in between – don’t worry you are far more than enough!!!

**You’re the rockstar now – Full Stack Web Developer!!**

**Blog and Other resources :**

* [**https://nodejs.org/api/all.html**](https://nodejs.org/api/all.html)
* [**https://devdocs.io/node/**](https://devdocs.io/node/)
* [**https://www.tutorialspoint.com/nodejs/index.htm**](https://www.tutorialspoint.com/nodejs/index.htm)
* [**https://www.tutorialsteacher.com/nodejs/nodejs-tutorials**](https://www.tutorialsteacher.com/nodejs/nodejs-tutorials)
* **FreeCodeCamp**
* **Edureka**
* **SimpliLearn**
* **Traversy Media**

**Youtube Channels :**

* [**Web Dev Simplified**](https://www.youtube.com/c/WebDevSimplified?app=desktop)
* [**Zero to Mastery**](https://www.youtube.com/c/zerotomastery)
* [**Code With Harry**](https://www.youtube.com/@CodeWithHarry) **( Hindi )**
* [**Programming with Mosh.**](https://www.youtube.com/@programmingwithmosh/videos)
* [**JavaScript Mastery**](https://www.youtube.com/@programmingwithmosh/videos) **- MVP**
* [**codedamn**](https://www.youtube.com/@codedamn)
* [**Pepcoding**](https://www.youtube.com/@Pepcoding)
* [**DevTown**](https://www.youtube.com/@DevTownIndia/playlists)

**-------------------------------------------------------------------------------------------------**

**System Design Course :**

**Nowadays, many product based companies are asking for some basic system design interview questions – The following resources will help you to prepare the basic – intermediate level of SD interviews.**

It is fun learning so, be casual while watching these playlists – carry less attention it’s ok even when you are travelling somewhere they’ll teach you the wonderful things in the application’s world. Happy learning!!

* [System Design Primer.](https://www.youtube.com/watch?v=FSR1s2b-l_I&list=PLTCrU9sGyburBw9wNOHebv9SjlE4Elv5a)
* [Gaurav Sen’s playlist on System Design Interviews.](https://www.youtube.com/watch?v=xpDnVSmNFX0&list=PLMCXHnjXnTnvo6alSjVkgxV-VH6EPyvoX)
* [Rachit Jain’s playlist includes Advanced topics on System Design.](https://www.youtube.com/watch?v=kZZvGuIDKnw&list=PLfBJlB6T2eOukvc2lrkAbeZBqUS94ji1r)
* [Miscellaneous.](https://www.youtube.com/c/SystemDesignInterview/videos)

If any links among are not working or any doubts related to the concepts – students can send you query as mail to this Gmail.

If there are any mistakes in the doc, **please** let us know through the Gmail so, that the updated version of docx will be uploaded in GitHub repository.

* Please give any suggestions ( on necessary changes ) so that I can update them in next version of this docx – which will be uploaded in github repository.

**Gmail :** [**Cp3helpdesk@gmail.com**](mailto:Cp3helpdesk@gmail.com)

**Github – Profile :** [**https://github.com/fs-aa**](https://github.com/fs-aa)

**Productivity Hacks :**

* In YouTube the maximum video speed is 2X – to increase more than 2X ( if needed ) add [**Video Speed Controller extension**](https://chrome.google.com/webstore/detail/video-speed-controller/nffaoalbilbmmfgbnbgppjihopabppdk?hl=en) to the chrome browser.
* As there **lot of advertisements** while using YouTube
* This chrome extension [**Ublock-Origin**](https://chrome.google.com/webstore/detail/ublock-origin/cjpalhdlnbpafiamejdnhcphjbkeiagm?hl=en)will block the advertisements.
* In many YouTube videos – there are sponsorships in between , so to avoid them use [**SponsorBlock Youtube**](https://chrome.google.com/webstore/detail/sponsorblock-for-youtube/mnjggcdmjocbbbhaepdhchncahnbgone?hl=en)chrome extension and disable all self-promotions, in dated sponsors.