



# What do I need to learn for front-end development?

One of the hardest things to do in modern web development is to figure out how to put everything together and how not to get fatigued by the sheer amount of technologies and methodologies out there.

I don't intend to be prescriptive but sometimes opinions sound like they are.

## What and how to choose

I got to this "roadmap" from [@javinpaul](#) through a question asked in Quora. The person asked naively, [Do I need to know everything in the front end road map posted by Hacker Noon website to become a front end developer?](#)

# Front-end

Learn the Basics

- Learn the basics
- Writing Semantic HTML
- Basic SEO
- Accessibility

HTML

CSS

Basics of JavaScript

- Syntax and Basic Constructs
- Learn DOM Manipulation
- Learn Fetch API / Ajax (XHR)
- ES6+ and modular JavaScript

Understand the concepts  
Hosting, Event Bubbling, Scope  
Prototype, Shadow DOM, strict,  
how browsers work, DNS, HTTP

- npm
- yarn

Package Managers

- SASS
- PostCSS
- Less

CSS Pre-processors

CSS Frameworks

- BEM
- OOCSS
- SMACSS

CSS Architecture



- Le
- Ma
- M
- L

npm impr  
in some f

PostCSS  
one. Go fo  
is still son  
if I was st

With modern front-e

Looking at the full roadmap is scary and this is just one part of a three-part document where they will outline everything you should learn or have available to learn if you want to work on the web.

It also troubles me how are items grouped together and what areas are not covered and why.

In later sections we'll discuss some alternative ways to look at what's available in ways that are not as intimidating as they are painted here.

## Starting from the bottom

These are some things that I consider necessary for front-end developers to be familiar with in order to do our jobs efficiently beyond knowing the tools we are required to use

### Git and Github

Github is the most popular Git hosting platform available and it will allow you to do many things other than hosting code. The fact that it is owned by Microsoft has turned a few developers away from Github to competitors.

### SSH

It allows you to login to another host remotely. This is necessary only if you need direct access to your server's shell to run configurations and modify the server in some way.

### HTTP / HTTPS

The HTTP protocol is the backbone of the web. Different developers need different levels of understanding of HTTPS and how they work. See Ilya Grigorik's [High Performance Browser Networking](#) for a good primer on the subject

### Basic Terminal Usage

Learn how to use the native terminal available to your Macintosh or Linux operating system

Learn how to use [Windows Subsystem for Linux](#), an equivalency layer that allows you to run a full Linux Operating system in Windows and run linux binaries from PowerShell and Windows programs from Linux

# Things that are nice to have

## Internationalization

If you are creating global applications which shows information in many different languages across the world then you should have a good knowledge of internationalization. We'll discuss more details of internationalization when we talk about HTML and CSS

## Performance

Front-End developers should be conversant with the basics of performance. How does the browser process the content of a page? What resources block and don't block rendering and how to help with performance using non-blocking resources as much as possible

# HTML: The core of the Web

HTML is perhaps the simplest of the three core skills for the web in terms of what we need to know in order to use it.

What is semantic HTML?

# CSS: Making it look pretty

# Javascript: Interactivity or dead weight?

I find the eternal argument of Javascript being required for modern web applications to be unrealistic and dangerous for the web as a platform.

I'm not saying "don't use frameworks at all" but "use frameworks wisely" and optimize them to make sure that your application will work well, no matter what.

Want to view a single image on [@imgur](#) (literally its only job)? Good luck! You gotta successfully download (354KB) and run (1.21MB) of client-side React in order to get your image requested as resource 110 of 553. What should have been an IMG element became... this. [pic.twitter.com/XkrbZcLQQg](https://pic.twitter.com/XkrbZcLQQg)

— Harry Roberts (@csswizardry) [October 19, 2019](#)

# Whatever you choose, use it well

I don't really care