

Web Development In 2018

A Practical Guide



Traversymedia.com

Before We Start...

- ✓ A guide to web development technologies and your options
 - ✓ Based on both fact and opinion
 - ✓ Look other places for guidance, not just this video
 - ✓ Decide what type of developer you want to be before anything
- [Full stack developer, front-end, web designer, back-end engineer & API creation]



Tools & Software

- ✓ **Text Editor** VSCode, Atom, Sublime Text
- ✓ **Good Browser** Chrome (preferred) or Firefox
- ✓ **Image Editing** Photoshop, GIMP, etc
- ✓ **Deployment Tools** FTP client, SSH tool
- ✓ **Cloud Storage** Optional Dropbox, Google Drive, etc



HTML / CSS

- ✓ Learn before ANYTHING else
- ✓ Building blocks of the web
- ✓ Every website uses both
- ✓ Easy & quick to learn
- ✓ Learn the core fundamentals & basics of Flex and/or Grid
- ✓ Responsive layouts are VERY important



Basic Vanilla JavaScript

- ✓ Basic JS [NO FRAMEWORKS!]
- ✓ Data types, functions, conditionals, loops, operators, etc
- ✓ DOM manipulation & events [WITHOUT using JQUERY!]
- ✓ Ajax / Fetch API / JSON
- ✓ ES2015+ (Arrow functions, promises, template strings)



Deploying a Basic Website

- ✓ Shared hosting [InMotion, Hostgator, etc]
- ✓ Learn the basics of cPanel [Creating email accounts, FTP, etc]
- ✓ Upload a site with FTP [FileZilla]
- ✓ Register a domain name and learn about DNS / Name Servers



Web Designer / Basic Front End Dev

- ✓ Build simple websites & UIs
- ✓ PSD / AI -> HTML / CSS
- ✓ Some dynamic UI with JavaScript
- ✓ Choose to be a freelancer or a very entry level position



Where To Next? [Depends]

- ✓ **HTML / CSS Frameworks** Bootstrap, Materialize, etc
- ✓ **Frontend JavaScript Framework** React, Angular
- ✓ **Server Side Language / Technology** Node.js, Python, etc
- ✓ **Database** MongoDB, MySQL, etc



HTML / CSS Frameworks [Choose One]

- ✓ **Twitter Bootstrap 4** General framework, most popular
- ✓ **Materialize CSS** Primarily a UI/UX framework
- ✓ **Bulma** Easy to learn syntax, No JS
- ✓ **Zurb Foundation** Alternative to Bootstrap
- ✓ **Skeleton** VERY light boilerplate



Front End JS Framework [Choose One]

- ✓ **React** Popular for startups, fast, Redux, JSX
- ✓ **Angular** Popular in enterprise, full featured, NgRx
- ✓ **Vue.js** Fast, light, easy to use, Vuex
- ✓ **Aurelia** Light, feels like vanilla JavaScript



Side Technologies You Should Learn

- ✓ **GIT** Version control
- ✓ **Basic Command Line** Folder navigation, file creation, etc
- ✓ **APIS / REST** Learn how REST APIs work
- ✓ **HTTP / SSL** HTTP requests, HTTPS
- ✓ **CSS Pre-Processor** Sass or Less
- ✓ **Webpack & Babel** Module loader & JS compiler



Server Side Technology [Choose One]

- ✓ **Node.js** Fast, scalable, and powerful
- ✓ **Python** Popular, rapid development, great integration
- ✓ **PHP** Practical, easy to deploy – Wordpress & Laravel
- ✓ **Ruby** Rapid development, strong community
- ✓ **C# & ASP.NET** Very powerful, Microsoft



Database [Choose One]

- ✓ **MongoDB** NoSQL, non-relational, suggested for Node.js
- ✓ **MySQL** Popular relational database
- ✓ **PostgreSQL** Powerful but a bit more difficult than MySQL
- ✓ **SQL Server** Microsoft's implementation of SQL
- ✓ **Oracle** Heavily used in enterprise
- ✓ **Firebase** Cloud database maintained by Google



Server Side Frameworks [Choose One]

- ✓ **JavaScript** Express, Hapi.js, Adonis, Loopback, Swagger
- ✓ **Python** Django, Flask, Web2py, Pylons
- ✓ **PHP** Laravel, Symfony, CodeIgniter, Yii2
- ✓ **Ruby** Ruby on Rails, Sinatra, Nitro
- ✓ **C#** .NET



Content Management Systems

- ✓ Wordpress, Drupal, Joomla [PHP]
- ✓ Great for clients
- ✓ Thousands of plugins / addons
- ✓ Fast development
- ✓ Can be limiting



Dev Ops & Deploying Applications

- ✓ Dedicated Server / VPS
- ✓ Cloud Hosting – Digital Ocean, Heroku, Amazon Web Services
- ✓ Cloud Storage – Amazon S3
- ✓ Working with SSH & command line
- ✓ Server maintenance & software updates



Mobile Applications [Choose One]

- ✓ React Native
- ✓ NativeScript
- ✓ Ionic
- ✓ PhoneGap / Cordova
- ✓ Xamarin



You Are a Full Stack Developer!

- ✓ Create simple to advanced web applications [Front & Back end]
 - ✓ Create secure REST APIs
 - ✓ Deploy & maintain applications
 - ✓ Administer databases
-
- You should now be able to get a very good job or run your own business



Biggest Changes From Last Year

- ✓ New technologies [Grid CSS, Fetch API, Materialize, NativeScript]
- ✓ More focus on full stack and REST APIs rather than server rendered views
- ✓ More focus on vanilla JavaScript in addition to frameworks
- ✓ Single Page Applications
- ✓ Python & C# over PHP



What Now?

- ✓ Focus on your career and whatever technologies your company uses
- ✓ Learn other languages / stacks in your spare time
- ✓ Stay up to date with trends & technologies



Follow Me

TWITTER:	Twitter.com/traversymedia
FACEBOOK:	Facebook.com/traversymedia
INSTAGRAM:	Instagram.com/traversymedia
DISCORD CHAT:	Discord.gg/traversymedia
LINKEDIN:	Linkedin.com/in/bradtraversy
GITHUB:	Github.com/bradtraversy