WebDevJourney

Jacopo Ferro - 28/06/22

Web Development learning journey for the next weeks.

Starting point

Extremely basic concepts of HTML, CSS, JS with no frameworks nor project experience.

I do have general CS background and know how to code in other languages, so I hope to pick up quickly on basic syntax of the new material.

Goals

Learn basics and more advance concepts of web application development.

Learn useful frameworks, good practices, workflows. Keep track of progress and versions.

Build an application for personal fun and learning.

Tentative Curriculum

This is my broad idea of what I think is a progressive curriculum for what I want to learn. It might change because as of now I don't have a good idea of what exactly is even required.

- HTML
- CSS
- JavaScript
- Front-End frameworks
- Full-Stack Application with Back-End

The idea is to first learn the syntax, the basic concepts, then consolidate it by applying it to small-medium projects along the way. Eventually introducing Back-End and frameworks that are actually used for development in industry.

Resources

Main resources I plan on using and other resources I stumbled upon while following the curriculum.
I will add links in the second category as I study.

Main structured Resources:

- #### Coursera HTML, CSS, JS (40h course)
- #### Youtube Playlist WebDev:
- - HTML (2h) - CSS (1.5h) - JS (3.5h)
- - JS Projects Examples (8.5h)
- - React (12h) - React Interactive Version
- #### Full-Stack Web Development with React Specialization:
- - Front-End Web UI frameworks and Tools: Bootstrap 4 (31h course)
- - Front-End Web Development with React (36h course)
- - Server-side Development with NodeJS, Express and MongoDB (48h course)

Extra noteworthy resources I discovered later:

- **Curriculum Related:**

  - - Higher Order Functions article
  - - JSDoc to document JavaScript programs found trough this article
  - - Import-Export Article (I got errors when texting offline)
  - - DOM - DOM docs - HTML DOM API - Examples
  - - Script/Interpreted vs Compiled languages - python example
  - - Fast React course (1h10')
  - - Starting a Node Server, (http-server for static is simplest, but had problems with it as it caches too long)
  - - CSS best practices (good revision too)
  - - Fix spread operator error in scrimba files
  - - Create React app without CRA - same
  - - VSCode TailwindCSS plugin bug fix

- **Big Picture:**

  - - Cool Back-End Overview Youtube video
  - - Cool Full Stack WebDev Roadmap Video
  - - Why React?
  - - Website builders: WordPress, Wix - WordPress .com vs .org
  - - Web Scraping/Interactions: Axiom vs Fetch - What is Puppeteer
  - - RSS feeds - Feedly (uses RSS feeds)
  - - TailwindCSS vs styled-components
  - - Wordpress vs Headless CMS
  - - What is JAMstack site
  - - Gatsby vs Next

- **Cool examples:**

  - - Tania Rascia cool personal site with tutorials and articles
  - - How to host Sanity Studio and automate build with GitHub Actions
  - - Josh Comeau
  - - React Scroll Parallax demo
  - - React Examples library

- - Inspo collections: [Brutalist](#) - [Pttrns](#) - [Awwwards](#)
  - - Digital Gardens: [Tom Critchlow Digital Garden](#) - [Andy Matuschak](#) - [Collection of Digital Gardens](#) - [Second Brains/Gardens](#) - [Digital Zen Garden](#) and [Gatsby Plugin](#) - [Another Gatsby plugin](#) - [This in chinese](#) + [source](#)

- ##### Frameworks pins & docs, official references:

  - [React Official Site, has nice tutorials](#)
  - [Codepen Online social coding environment](#)
  - [Typescript](#) - [MDX](#)
  - [Deno as JS/TS environment (Instead of Node)](#) - [Fresh as fullstack framework of Deno](#)
  - [http-server](#) - [Live Server](#)
  - React Frameworks: [Gatsby framework](#) - [Gatsby advanced starter](#) - [Create-React-App docs](#) - [Next.js](#) - [LIST of Site Generators on JAMstack](#)
  - Build and deploy: [Netflify to build and deploy](#) - [Vercel](#) - [Railway](#) - [Heroku (fullstack)](#)
  - [Figma for design](#)
  - Web interactions (scraping and more): [Axios as better Fetch](#) - [Cheerio for parsing HTML on server with Node.js](#) - [Puppeteer as cheerio on steroids, simulates browser](#)
  - Useful CSS base files: [Reset css](#) - [Normalize CSS](#) - [Primitive CSS](#)
  - CSS Frameworks/Libraries: [TailwindCSS](#) - [Tailwind UI kit](#) - [Headless UI](#) - [Heroicons](#) - [Styled Components](#) - [Chakra UI](#) - [Daisy UI](#) - [Tailwind Components](#) - [Flowbite (Tailwind)](#) - [One Page Love](#)
  - Animation and others: [Three.js for 3D](#) - [Rive for animation](#) - [Coolors to create/choose color palettes](#) - [React Spring](#) - [Framer Motion](#) - [Pattern Library](#) - [React Scroll Parallax](#)
  - Building on your own (only React or adding layers one at a time): [Babel for language definition (highlighting JSX)](#) - [Webpack](#) - [Parcel (nice)](#) - [Vite, module bundler like webpack](#)
  - CMS: [Sanity](#) - [Contentful](#) - [Strapi](#) - [Hygraph](#) - [LIST of Headless CMS on JAMstack](#)
  - [Measuring website performance, SEO](#)

Note: I don't want to clutter the resources section so I won't add every resource that is directly connected to the main ones (interactive versions of the youtube courses for example).

Consuming these extra resources might shows up as 'Other' in the WorkLog

Some of these things ended up becoming somewhat 'unproductive' rabbit-holes, I will try to select just a few and strike a balance between pinning for reference in the future and adding only the useful things that add side info while doing the curriculum.

Workload

Expected Workload in actual focused deliberate practice hours:

Set-up, docs, progress tracking

- Ignored for simplicity, like this *README* or finding resources

Basics and Front-End

- **HTML:** 3h
- **CSS:** 2h
- **JS:** 4.5h
- **Front-End Projects and Practice:** 15h
- **React:** 20h

Specialization:

- **Course 1:** 30h

- **Course 2:** 30h
- **Course 3:** 45h

Personal Projects and Practice:

- I will try to carve out time for personal portfolio projects while following the curriculum but this is the one final point that can expand indefinitely once I know how to build.

Total net expected time: *150.5h*

Day to Day

I track my net working ours with timers and remove any break or pause, hence
5-6h of focused work might mean much more dedicated time to the tasks.

A realistic estimate including days off and holidays during this summer is 20-25 working days in July and same in August.

During these two months I have other tasks and projects to do so I suppose it will be a challenge to keep the pace of
5.5h/day (4h webdev + 1.5h other coding) for 40 days or 4.5h/day for 50 days.

I don't know if I will update this *README* often or for simplicity I will just privately keep track of my progress.

Just Kidding here's a tentative table I will update with various logs:

WorkLog

Log	Topic	Net Hours	Notes
28/06/22	Setup	5h	git, atom, resources, *README*
28/06/22	HTML	2h40'	followed video HTML
29/06/22	CSS	2h	video CSS, then interactive version
29/06/22	Other	1h	articles and peak at new resources
30/06/22	Other	1h30'	
30/06/22	CSS	2h30'	CSS basics and Grid
01/07/22	CSS & Other	1h15' + 1h15'	CSS flexbox & considering Vue.js instead of React
01/07/22	JS	1h	Setup Atom and start course
03/07/22	JS	7h	JS course, HOF, JSDoc
04/07/22	JS Projects	1h40'	Vanilla JS Color Flipper, Counter
04/07/22	Other	2h30'	Site builders, DOM, React & other resources
04/07/22	React	1h50'	Cut time on Vanilla HTML/CSS/JS projects, dive into React
05/07/22	React, Other	3h30'	Discovered and added a bunch of side resources
06/07/22	Other	1h	Cleaned Resources

Log	Topic	Net Hours	Notes
06/07/22	React	2h'30	Still absorbing basic resources and Setup react
08/07/22	React	1h	More setup and course
09/07/22	React	5h	Course + Trying to solve problems with import/export!
09/07/22	React	1h	stick with create-react-app
10/07/22	CSS, React	5h15'	React Info_Site, Figma, CSS practice
10/07/22	React, Other	1h30'	Use Scrimba files, less weight than create-react-app
11/07/22	React	1h45'	AirBnB Clone using Scrimba code, will follow course faster
11/07/22	React	1h	SetState on vars, on objects, on arrays
12/07/22	React, Other	4h	More react, modded light setup with webpack
13/07/22	React, CSS, Other	2h10'	Tailwind CSS, webpack setup, babel, other cool stuff
14/07/22	React, CSS, Other	7h	Understanding frameworks, TailwindCSS
15/07/22	React, Other		Frameworks 'fatigue'! Many hours understanding big picture
16/07/22	React		Sidetracking: experimenting with Gatsby, TailwindCSS
17/07/22			
18/07/22			
19/07/22			

Other refers to exploration regarding curriculum, further articles, project ideas and in general lateral activities and learning that are parallel to the main focus but would be harder to keep track precisely.
Examples: markdown and environment for the *README*, articles about tech stacks, docs of Frameworks/libraries and more.

Authors

- @jacopoferro