

# Top JavaScript Frameworks and Topics to Learn in 2020 and the New Decade



Eric Elliott

Jan 3 · 14 min read



If you want to land a great JavaScript job or catch up on important tech for 2020 and important technologies in the new decade, this post is for you. The point of this post is not to tell you which tech stacks and frameworks are the “best” or most loved or most popular — but to shed some light on which ones give you the best odds of landing a great job in 2020 and beyond.

We’re going to look at data from a variety of sources:

- Google Search trends by topic
- State of JS Survey
- Stack Overflow Survey
- Npm downloads
- Aggregated job postings

None of these metrics are perfect, but for our purposes, the npm download counts and job postings get the most weight, and when we look at the metrics in aggregate, they paint a clear and consistent picture of the JavaScript technology landscape and trends. Is it a good idea to select a framework to learn based on these metrics? *That depends on your goal.*

Since we have a clear goal — learning ROI for job seekers — that makes the task much easier than trying to tell you what's best for everyone. While there is no such thing as best for all purposes, it's pretty easy to objectively define what will give you the *best odds* of achieving a specific, concrete goal. Let's look at some data.

## First, Learn JavaScript

Before you worry too much about tech stacks, learn JavaScript and how to compose software with it. If you can't explain what function composition, object composition, and modules are, start here. All software development is composition: We take a large, complex problem and break it down into smaller problems that we can solve with the building blocks of software: Functions, objects, modules, and so on. We assemble those solutions to form our applications. Start 2020 with a good understanding of JavaScript and software composition.

## React Dominated 2019, and Will Probably Dominate 2020



I like the npm downloads metric because it gives a pretty good indication of a framework's active usage\*. Frameworks used commonly in industry will score uncommonly high on downloads because users are running `npm install` on their local machines.

*Note: jQuery is excluded here because many jQuery projects are legacy projects which do not use npm, so it would be severely undercounted, anyway. Svelte is included to provide some insight on its relative position in the market, and explain its absence in the rest of the charts — there is not yet enough data for Svelte to make a meaningful showing in the charts. For example, it is not yet an available topic in the Google Search Trends.*

## Aggregated Job Postings

This chart counts currently open job postings mentioning a particular framework.\*

Dec 2019 Job Board Listings by UI Framework — Source

React strengthened its lead vs last year, while both Angular and jQuery lost ground to Vue. Here's a pie chart showing the relative job market share for each framework:

### Top Framework Job Market Share

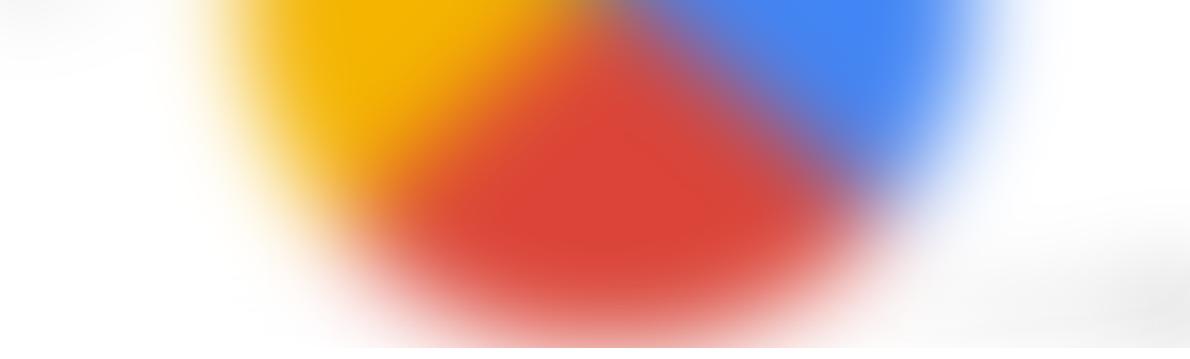
Average JavaScript developer salary increased again in 2019, from \$111k/year to \$114k/year (source).

*\*Methodology: Job searches were conducted on Indeed.com. To weed out false positives, I paired searches with the keyword “software” to strengthen the chance of relevance, and then multiplied by ~1.5 (roughly the difference between programming job listings that use the word “software” and those that don’t.) All SERPS were sorted by date and spot checked for relevance. The resulting figures aren’t 100% accurate, but they’re good enough for the relative approximations used in this article.*

## Google Search Trends

Source

As you may expect, search interest\* is somewhat reflective of the job market share, with some interesting differences. We can clearly see a waning interest in jQuery between 2017 and today, along with strong growth in Vue.js search interest. Here are the search shares:



As with the job postings, React has a strong lead, attracting 36% of the search volume for front end frameworks, followed by Angular at about 27% and jQuery at 25%. Search interest in Vue.js is larger than its job market share by a healthy margin, but this data generally agrees with the job market data in rankings. It looks like we have successful corroboration.

**\*Methodology:** *For all terms, data is collected by topic, rather than search term in order to weed out false positives from irrelevant keyword matches.*

## Frameworks to Watch in the Future

More jobs are looking for skills with React than for skills with any of the other popular, but less widely used frameworks such as Svelte or Vue (which both have *very high satisfaction ratings*, but comparably far smaller industry adoption).

**It may be cool to learn Svelte or Vue — but if your goal is to get a job, you'll have better odds if you learn React first.**

That said, both Svelte and Vue scored very high on user satisfaction in the State of JavaScript 2019 survey. In the past, React's high user satisfaction scores predicted rapid gains vs Angular, which was dominating at the time, but had comparatively low satisfaction scores.

In 2019, React ranked highest in user satisfaction with 89% users satisfied. Runners up include Svelte (88%) and Vue.js (87% — down from 91% last year). It's unlikely that Svelte or Vue will steal enough React users to make a dent, but there are still lots of users using Angular and jQuery who could defect to Svelte or Vue, driving further strong growth for both frameworks in 2020.

It's a strong bet that having a mastery of React will increase your odds of finding and retaining a great job in 2020.

## TypeScript vs JavaScript

There is no question TypeScript has grown very rapidly over the course of the last few years, and according to the State of JavaScript survey, 89% of TypeScript users would use it again, and 66% of survey respondents either use TypeScript, or are interested in using TypeScript (down slightly from 71% least year).

But while interest is certainly strong and usage seems to be growing quickly, experience with TypeScript is not yet in strong demand on the job market. Only ~7% of JavaScript job openings mention TypeScript in the job description (source). This is probably undercounting TypeScript jobs a little because hiring managers expect JavaScript developers to be capable of picking up TypeScript without much trouble, so there's a chance they won't bother to mention TypeScript in their job postings.



I stand by my assessment that the TypeScript language may have a low or even negative return on investment. It could hurt rather than improve your productivity, and if you're already using great bug prevention measures such as TDD, code review, and

design review, coding in TypeScript is unlikely to provide a significant bug reduction benefit.

That said, there is a lot to love about TypeScript, and you certainly should not be afraid of it, or turn down a job because they use TypeScript instead of JavaScript. Because TypeScript is a superset of JavaScript, onboarding from JavaScript to TypeScript is not as challenging as learning an entirely different language.

You probably don't need to learn the TypeScript language to get an edge on the job market competition in 2020, but the TypeScript engine is quite useful, even for standard JavaScript.

I use it every day to provide intellisense for standard JavaScript using Visual Studio Code. That intellisense can even be enhanced using JSDoc (which the TypeScript engine understands and interprets as it would interpret type annotations) or external d.ts files, and VS Code will automatically acquire TypeScript definitions for the modules you use.

*Note: I've been enjoying similar benefits for several years using TernJS and Atom, but that combination lacks the maintenance and community support of TypeScript engine + VS Code.*

If you haven't tried Visual Studio Code yet, you may want to start there. BTW, VS Code dominates the JavaScript IDE market among State of JS respondents, with 57% market share (followed by WebStorm, with 14% market share).

## Data Management

Redux still has a strong lead in the state manager race, but GraphQL and Apollo are gaining in satisfaction and interest. Expect to see continued growth of GraphQL in 2020.

That said, I still think there's a strong case for Redux state management even if you use GraphQL, and I think everyone would benefit from learning Redux even if they don't use the framework.

GraphQL is a query language that has become popular in recent years. It has simple syntax, but because it's used across the full stack and learning it entails learning how to hook it up to your data storage layer, it can be intimidating to integrate — but well worth the effort.

GraphQL is not going to completely replace JSON REST APIs any time soon, but it grew even faster in 2019, and I expect strong growth to continue into 2020.

## Back End

Express is the dominant Node framework, with very high satisfaction and no serious challengers going into 2020, but with the rise of serverless, I expect to see Express dominance slip as the new decade unfolds.

Next.js is a full stack React framework which was initially built on top of Express, but has since switched from Express to **serverless** and static optimization and it's growing like crazy.

When we switched EricElliottJS.com from Express to serverless Next.js, it cut our hosting bill 90% and sped up page loads. I can't recommend it strongly enough. We're now using Next.js and Zeit hosting for all our apps.

## Testing

Jest and Cypress are most popular for unit and functional testing, respectively, but I have a strong preference for RITEWay and TestCafe. RITEWay is unit testing zen: A testing framework which always answers the 5 questions every unit test must answer.

TestCafe is a functional testing tool which, like Cypress, does not require you to struggle with Selenium, but has better cross browser support and a cool test recorder/IDE. I strongly recommend QA teams check out TestCafe.

## Functional Tools

Lodash, Ramda, Immer, Redux, and RxJS are the functional tools I frequently use.

Ramda grew a lot in 2019, and offers some tools that are not features of Lodash, including lenses and transducers. RxJS offers transducer style functionality with its pipeable operators.

Immer makes it easy to operate on JavaScript objects without mutating them. Redux is a state management tool.

## The Tech Landscape in 2020

There are several emerging technologies which are set to make a dramatic impact on human life over the course of the next decade. Here's a rundown of the current state of the art in each:

### AI

AI may be the most transformational technology ever developed. Various scholars and opinionated billionaires have described AI as a miraculous utopia, or the end of humanity.

It certainly could be the end of humanity as we know it, but it could be the beginning of a beautiful collaboration between humans and machines. We're only beginning to catch shadowy glimpses of what AI could become. Some of it scary (like deep fakes) and some of it extremely useful (like Adobe Premiere's new auto-reframe for video editing). The truth so far is that AI is both incredibly scary and incredibly useful. Some scary things will continue to come out of AI, but the potential to improve human life with AI is tremendous. Pandora's box is already open, and there's no closing it again, so it is my hope that you'll use it well to solve important problems and improve lives.

There were many important breakthroughs in the field of AI in 2019. In recent years, research teams have been competing for superhuman performance in a variety of video games, but some classes of games remain challenging for AI, including long term strategy games such as StarCraft II. DeepMind's AlphaStar program has achieved grandmaster level playing on the official Battle.net servers using the same resources human players have access to. StarCraft is a complex game involving long-term strategic thinking once thought to be beyond the capabilities of AI. Strategic thinking represents a major milestone on the path to general AI.

If these games seem like a silly use of resources, consider that projects like DeepMind have also made incredible advancements in text-to-speech, which can give AI agents and personal assistants more natural voices, but also importantly, give voices back to humans who have lost them.

AI is also being used to diagnose cancer, predict protein folding (which could be used to learn about disease and discover new medicines), restore limb control for the disabled, generate photo realistic images, etc. (Source)

**Self driving cars** deserve their own category in transformational AI technology. Big players in that ecosystem include Cruise, Uber, Waymo, Nuro, Aurora, Ford, etc. It's still early days, and over-optimism has caused real problems for the industry, but I have no doubt we'll see superhuman self-driving capabilities inside the 2020s. I made a prediction about self driving in 2015:

**“By 2045, a human driving a car will look like a horse pulling a buggy.”**

5 years later, I think I may have underestimated how fast self-driving cars would hit the market. Waymo vehicles drove over 1 million miles in 2018, and 24 US cities had self driving cars on public roads. Today, I'm predicting full level 5 autonomy in multiple car models from multiple manufacturers by 2025. Keep your eyes on Tesla, GM, Ford, and BMW.

AI research paper publication has grown 10x over the past decade, and now that AI is putting practical, useful tools into the hands of mainstream users, expect that trend to continue into the new decade.

We're already starting to see AI do things that most people would not have believed possible just a few years ago. Expect AI to work even more stunning miracles in the 2020s.

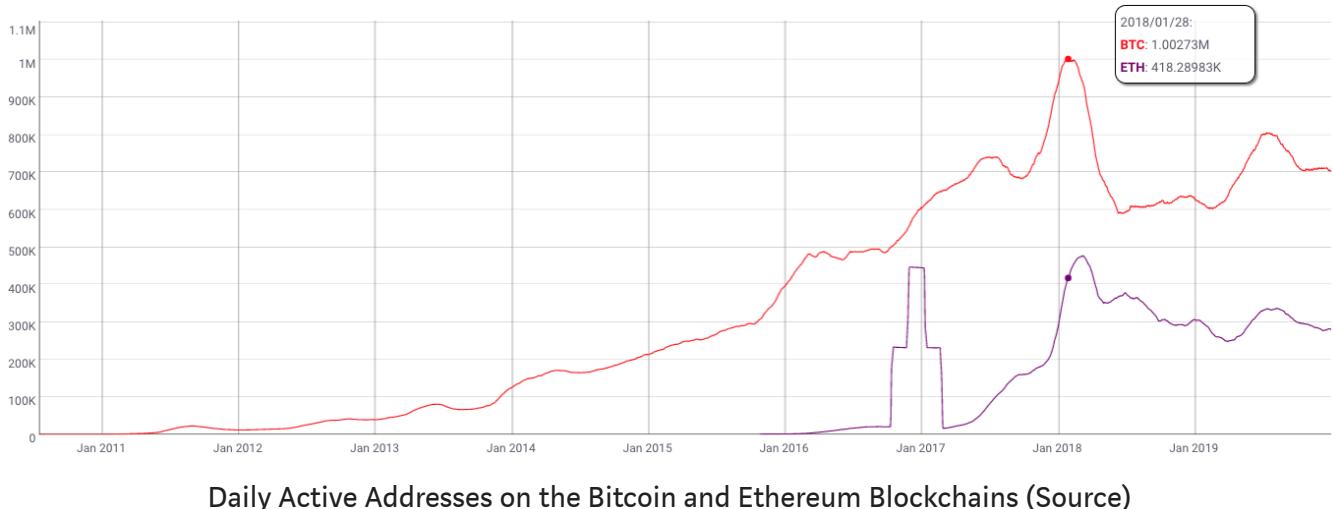
## Blockchains and Cryptocurrencies

2019 was a monumental year in the world of blockchains and cryptocurrencies. Blockchain technology is beginning to break into the mainstream. I'm expecting even more growth and mainstream adoption in 2020.

**DeFi** exploded, enabling user friendly, permissionless, and bankless borrowing and lending. There is currently \$650 million locked in DeFi contracts (including \$450 million in bankless loans), and 2019 was the first year that this technology was available. As more people invest in cryptocurrency, I predict a lot more crypto-backed DeFi loans in the future.

People can access liquidity without sacrificing the opportunity cost of their crypto investments, or use DeFi loans to invest in crypto with leverage (increasing the risk/reward of their investment).

Crypto and blockchain technology is breaking through to the mainstream. Daily active addresses for Bitcoin and Ethereum peaked in January 2018, followed by a year of waning interest due to the down market (after every 10x climb, crypto prices fall back dramatically before climbing 10x higher than the previous peak in the next cycle).



- **Square's Cash App has over 10 million Android downloads** — users can buy, sell, send and receive Bitcoin in a simple app with great UX.
- **Coinbase has over 10 million Android downloads.**

- **Brave** launched **built-in Ethereum wallet** and went from 8.7 million monthly active users in October to **10.4 million MAU** at the end of November.
- **20 million** video game fans are using the Theta network for decentralized streaming thanks to Sliver.tv and DLive.

My favorite development of 2019 was the introduction of **Fortmatic**. Whether you're building a crypto app or not, you can replace your user authentication with Fortmatic and improve security, break free of vendor lock-in, and add capabilities such as end-to-end encryption, signed transactions, etc.

If you're building a new app in 2020, consider using Fortmatic instead of rolling your own username/password auth (which nobody should be doing in 2020), or delegating auth to Google, Facebook, etc.

I wrote an article discussing how to leverage crypto technology to improve app authentication and security. I am looking forward to seeing mainstream apps adopt technologies like Fortmatic in 2020.

Ethereum easily won the smart contract platform race in 2019. I predict it will continue to dominate in 2020.

## Extended Reality (XR)

I've been writing about the promise of AR every year since 2015, and dreaming about it since I was a kid. In the future, XR will replace cell phones. It's only a matter of time. How much time is still an open question, but it looks like the shift could come in the next decade. It could even start within the next 5 years.

As the lines have blurred between VR and AR, the industry is now referring to the whole field as XR (eXtended Reality). ARKit and ARCore (Apple and Google's Augmented Reality SDKs) have grown a lot since their introduction. Monthly active users grew from 47 million to 150 million between 2017 and May 2019.

WebAR gained a foothold in 2019, and it's likely to continue to grow in 2020. If you're curious check out AR.js, React 360, or Viro React.

There have been many advancements in hardware. Hololens 1 was about \$5k when the developer edition was first released in 2016. What's happened since then?

Hololens 2 is still \$3,500, and they're not ready for consumers, yet. There are still issues with clipping, SLAM (jitter and drift), field of view, etc. And you don't want to wear these things all day. But thanks to the \$99/month pricing, getting your hands on a developer kit may be a possibility — even if you have to sit on the waiting list for a while.

## Drones

About 2 million drones have been sold in the United States, and they have already transformed construction, geographic surveys, mapping, film, aerial photography, farming, environmental science, and fun. There's a good chance commercial drone deliveries will finally be a thing in the United States in 2020.

FPV Drone Racing is my new favorite spectator sport, and the Drone Racing League live streams races. They also have a carefully engineered simulator, if you want to get an idea of what it's like.

All this new drone activity requires a lot of software, including image processing, drone mapping software, flight planning for deliveries, telemetry recording and processing, analytics, industry application integrations, etc.

There's also a lot of overlap with AI — self flying drones require image processing, collision avoidance and so on.

## Conclusion

The 2010s gave us a lot of transformative technologies, many of which we're already taking for granted, but had a big impact:

- Instagram
- Spotify
- Uber
- Virtual assistants
- Wireless earbuds
- Mainstream VR
- Augmented reality (as used in the breakout hit game, “Pokémon GO”)
- Affordable Teslas
- The sharing economy (cars, scooters, bikes)
- Mainstream tablets
- Mainstream biometrics (touch ID, face ID)
- Crowdfunding

I can't wait to see what you build in the new decade.

Make some magic.

## Next Steps

Learn React, Redux, Next.js, TDD and more on EricElliottJS.com. Access a treasure trove of video lessons and interactive code exercises for members.

• • •

**Eric Elliott** is the author of the books, “Composing Software” and “Programming JavaScript Applications”. As co-founder of EricElliottJS.com and DevAnywhere.io, he teaches developers essential software development skills. He builds and advises development teams for crypto projects, and has contributed to software experiences for **Adobe Systems, Zumba Fitness, The Wall Street Journal, ESPN, BBC, and top recording artists including Usher, Frank Ocean, Metallica, and many more.**

He enjoys a remote lifestyle with the most beautiful woman in the world.

Thanks to JS\_Cheerleader.

JavaScript   Technology   React   Future   Software Development

About   Help   Legal

Get the Medium app

