What is a front-end developer

I’m an old school developer. The first website I made dates back to 1995 when a friend and I built our school’s first website for independent study credit.

The web has come a long way from those ealy days but with all the good has come some separation and arguments that never seem to end. The latest one seems to be about the role of a front-end developer and what is front-end development as a discipline.

The rest of the post will work on the following assumptions:

- Front-end developers should be equally proficient with HTML, CSS and (vanilla) Javascript. I don’t mean they should be experts but at least enough to formulate a good question for Stack Overflow or how to search MDN
- Javascript knowledge should not be tied to particular frameworks or libraries. We should separate basic Javascript used for interactivity from the tooling and frameworks we use create and package the content
- CSS Knowledge should not be tied to frameworks or depend on forward looking technologies

Javascript, since it’s inception in 1995, has become the must have, must use tool for web developers used in all kinds of projects that touch the web in some capacity.

From serverless, to CSS in JS to bundling, all newer technologies now revolve around Javascript and I think that’s where the problem starts but not where it ends.

Javascript, CSS, UX, oh my! or How to be a front-end developers

Over the years the job of creating web content has become more and more specialized. It’s not enough to know the basics anymore but we’ve added piles of stuff that may or may not be strictly front end to the mix.

I think that the first thing we need to do as front end developers is create a basic set of competences that we all agree front-end developers should have... I know it’s opening a can of works in wanting all to agree what are the core competencies for the front end but I think it essential that we all “play from the same playbook”.

I did a search on Indeed.com for the words “front end developer” and was somewhat surprised at the specificity of what most of these jobs where looking for in a front end person and also how little some of it has to do with the crafting of a web application front-end.

Some of the things that bothered me about what I found begin with the fact that there is no uniform description of what makes for a front-end web developer. I would exppect some basic competencies, for example, how many years of HTML, CSS and Javascript do you expect a front-end developer to have for a position in your company?

I would also like to see common-sense and realistics expectations of experience based on how old and/or complex a given framework is? How do recruiters, and the people who tell HR what they need for a given position, expect to get quality applicants if they want a React developer with 7 years of experience (in a framework that has been publically available for 6)?

Where do we draw the line?

The arguments over front-end versus back-end versus full stack are not new; it’s the tools that have changed. Whether we like it or not our front end experiences have become Javascript heavy and we’re paying the price for it.

But that begs two questions:

When you work in large Javascript codebases that produce user-facing content programmatically, are you a front-end developer or a programmer?

Conversely if you’re great at CSS but may not know Javascript or HTML that well, are you really a front-end developer? Do you create the markup you style? How do you create the interactions that require Javascript?

One of the comments in Chris’ article that called my attention was Steven Davis’:

> **I think we need to move away from the term myself. PHP/ASP developers are dying off. Those people are moving into mobile development or JS. We should split old school "front end" into UX Engineers and Javascript Engineers IMO. They are different mindsets. Most people are not amazing at both JS and CSS. It's usually either/or. Let UX Engineers work closely with UX/Design to create great interations, prototypes, interaction events, etc and let JS Engineers handle all the data part. Just my 2 cents. So sick of being great at CSS but being forced into JS. I'm not a programmer!**
>
> What makes a good front-end developer? comment

It left me with more questions than it answered. If we were to break things like the comment suggests… How do you prototype interactions for your web content? What role do we expect UX engineers to take in the coding of the prototype? Are you expecting to have a Javascript engineer as part of your UX team? Or are you just handling static mocks with instructions for the JS engineers to work with?

But along with those questions comes a concern about the community and how we're onboarding new people ass front-end developers and what do we expect from them.

How do we move forward as a community?

In 2012 Rebeca Murphey wrote A baseline for front-end developers as a minimum of what people should know to work in front-end development at the time it was written.

I went back to the article and was surprised at how well it holds up 7 years later; the tools may have changed but the sentiment that we should all have a common base to start the conversation definitely has not.

We should all come to an agreement on what do we want the basics to be. What are the basics that someone should know for a front-end developer position?

I have opinions and have written about them elsewhere. But I'm one voice and one opinion, would love to hear more about what other people think and how can we build a dialogue about this.

Links and resources

- Shoptalk podcast How to be a front-end developer series: episodes 331 - 338
- The Great Divide
- What makes a good front-end developer?
- Is front-end development having an identity crisis?
- Job description tweet
- How Frontend Developers Can Help To Bridge The Gap Between Designers And Developers