Nouveau message

À

What is a Full Stack developer?

Is it reasonable to expect mere mortals to have mastery over every facet of the development stack? Probably not, but Facebook can ask for it. I was told at OSCON by a Facebook employee that they only hire 'Full Stack' developers. Well, what does that mean?

To me, a Full Stack Developer is someone with familiarity in each layer, if not mastery in many and a genuine interest in all software technology.

Good developers who are familiar with the entire stack know how to make life easier for those around them. This is why I'm so against silos in the work place. Sure, politics and communication challenges get in the way in large organizations. I think the point Facebook is going for with their hiring policy is, if smart people use their heads and their hearts, a better product gets built in less time.

LAYERS OF THE FULL STACK:

1. **Server, Network, and Hosting Environment.**
   A. This involves understanding what can break and why, taking no resource for granted.
   B. Appropriate use of the file system, cloud storage, network resources, and an understanding of data redundancy and availability is necessary.
   C. How does the application scale given the hardware constraints?
   D. What about multi-threading and race conditions? Guess what, you won't see those on your development machine, but they can and do happen in the real world.
   E. Full stack developers can work side by side with DevOps. The system should provide useful error messages and logging capabilities. DevOps will see the messages before you will, so make them count.
2. **Data Modeling**
   A. If the data model is flawed, the business logic and higher layers start to need strange (ugly) code to compensate for corner cases the data model doesn't cover.
   B. Full stack developers know how to create a reasonably normalized relational model, complete with foreign keys, indexes, views, lookup tables, etc.
   C. Full stack developers are familiar with the concept of non-relational data stores and understand where they shine over relational data stores.
3. **Business Logic**
   A. The heart of the value the application provides.
   B. Solid object oriented skills are needed here.

      C. Frameworks might be needed here as well.

4. **API layer / Action Layer / MVC**

    A. How the outside world operates against the business logic and data model.

    B. Frameworks at this level should be used heavily.

    C. Full stack developers have the ability to write clear, consistent, simple to use interfaces. The heights to which some APIs are convoluted repel me.

5. **User Interface**

    A. Full stack developers: a) understand how to create a readable layout, or b) acknowledge they need help from artists and graphic designers. Either way, implementing a good visual design is key.

    B. Can include mastery of HTML5 / CSS.

    C. JavaScript is the up and coming language of the future and lots of exciting work is being done in the JavaScript world (node, backbone, knockout...)

6. **User Experience**

    A. Full stack developers appreciate that users just want things to work.

    B. A good system doesn't give its users carpal tunnel syndrome or sore eyes. A full stack developer can step back and look at a process that needs 8 clicks and 3 steps, and get it down to one click.

    C. Full stack developers write useful error messages. If something breaks, be apologetic about it. Sometimes programmers inadvertently write error messages that can make people feel stupid.

7. **Understanding what the customer and the business need.**

    A. Now we are blurring into the line of architect, but that is too much of a hands off role.

    B. Full stack developers have a grasp of what is going on in the field when the customer uses the software. They also have a grasp of the business.

Other Pieces of the Puzzle:

1. Ability to write quality unit tests. By the way, even JavaScript can have unit tests these days.
2. Understanding of repeatable automated processes for building the application, testing it, documenting it, and deploying it at scale.
3. An awareness of security concerns is important, as each layer presents its own possible vulnerabilities.

Closing Thoughts:

It is very bad practice to tightly couple code to a specific implementation (library, OS, hardware, etc). Just because a full stack developer understands the entire spectrum doesn't mean they have license to take shortcuts. Well, actually they do if it is a build and throw away prototype.

Technology start-ups need full stack developers for their versatility! However, as an organization matures, it needs more and more focused skills.

I'm not sure you can call yourself a full stack developer until you have worked in multiple languages, platforms, and even industries in your professional career. Full stack goes beyond a 'senior engineer', as it is along the same lines as a polyglot programmer but with a higher view of all the connecting pieces. Note that on my list, only items 3-5 involve writing code.

GET EMAIL UPDATES ON THE PROGRESS OF MY WRITING

I'm working on a *[core dump, implant, book]* that contains timeless lessons in software development. I wish every developer knew this stuff, especially developers just starting out!

your_user@awesome.com

Subscribe