MICHAEL DAVIS

SOFTWARE ENGINEER TEAM LEAD @ STRATOS CLOUD

TAGLINE

"FAST. GOOD. CHEAP. Choose any two"

CHARACTER CUES

PET PEEVE

"I like to actually get work done when I'm at work. One thing I absolutely despise is having to go back and forth over and over again to 'clarify' things that should have been clear from day 1. It's a waste of my time and I don't have time to waste."

OTHER PEOPLE SAY

"Wicked smart, he's an excellent engineer. I would hire him if I had any software that needed building and could afford his rate."
"Michael is a top notch engineer, though at times he can be a little rough around the edges, especially if someone is having trouble understanding what he's saying, which is basically every non-engineer."

BUSINESS DOMAIN INFO

Michael is an expert engineer and has been working in the field for nearly 10 years. He's loved computers since before he could even remember and chose to study that in college. After college he dove into the world of frontend software development. Now, Michael is the authority among his team on front end architecture. If any front-end engineers on his team have questions or concerns they come to him. He doesn't mind being a leader but would sometimes rather just be left alone so he can finish work and go home. Nonetheless, he finds himself constantly interrupted by engineers or designers regarding requirement changes and clarifications. He wonders why designers can't just design things right the first time, or at least provide better requirements but he's kind of given up hopes that that dynamic would ever change.

PERSONAL INFORMATION

Occupation: Front End Software Engineer
Location: Austin, TX
Age: 30
Sex: Male
Highest Edu: BS from Stanford
Study: Computer Science

HARDWARE/SOFTWARE

PC: Latest MacBook Pro 15"
Phone: iPhone Xs 256gb with ∞ data
Tablet: iPad mini 4, "mostly for kindle"

GROKKABILITY

Fav Books: Algorithms to Live By
Most Opened App: Reddit
Hobbies: Homebrewing, Board Games

IN MICHAEL'S OWN WORDS

I've loved computers since before I could remember. My first computer was a hand-me-down from my parents that I got when I was about 8 years old. I still remember it as a large, heavy, and beige Packard Bell computer. Yes, that's the "Packard" from Hewlett Packard, that's how long ago this started. I played a lot of Oregon Trail on that machine, that's about all it was capable of doing but eventually I had to level up and that's when my interest in computers really started to take off. I built my first computer at 13 years old and the rest is history. Somewhere in that history is a BA at Stanford in Computer Science as well and now I'm ten years seasoned as a professional software engineer.

During my first couple years as a software engineer I learned a lot on job, a lot that they don't teach you in college. They don't teach you about the frustration you get that stems from managers and other engineers who simply can't communicate. But I digress, I found that in software engineering this is just the way things are. Management often tries to estimate projects that will never get completed in time. They really ought a read "The Mythical Man Month" and tack a couple weeks onto every estimate they give because it's inevitable that what they ask for isn't really what they want, what they want isn't what they need and what we all agree upon and end up with is nothing like anyone expected in the first place. That's software engineering for ya!

It gets better though. Once you're in the game for a while and you're on a team that understands what software engineering is really like, then everyone is one the same page and everyone realizes that everything is just sort of a mess all the time. It doesn't make the work any easier but it makes it less frustrating. It give's everyone a "we're all in this together" sort of vibe that we can all really resonate with. Then, as long as we can manage to hit close to our goals/KPIs we're golden.

I think most of these problems stem from poor communication, honestly. Engineers aren't the best at communicating so you can't really expect to team them up with a bunch of designer's and business men and expect a dream team to form. And to add to that, many engineers aren't really even self-aware enough to notice their poor communication. I like to see myself as a fairly good communicator but maybe i'm just part of the aforementioned group. Over time I've been part of teams that try to fix this communication gap, often times either with apps or team building exercises of some sort but many times it doesn't help and if it does it's only temporary.

Oh, and don't get me started on designers. I've had my fair share of run ins with those types. Often times it seems like they can't make up their minds and if they can do that then they can't communicate what they want and if they can do that then they're designs just plain suck. If your designer gets passed three of those checks, then don't let them leave because you have a unicorn on your team. Unfortunately for me, every designer I've worked usually exhibits at least one of those three traits. Sometimes they exhibit all three; boy is that fun. But I've grown used to it, it's just something you have to adapt to once you start your career in software engineering and once you get used to it, it really isn't that bad.