Non-Technical Interview Guide

**Q: Tell me about yourself**

* The most common question! Try to practice a 30 second introduction to yourself. Mention your school, recent experience, skills, and what you’re interested in. For example,
  + “I’m (your name), I go to/went to (your school). Recently, I worked at (recent experience) and my skills include (skill 1), (skill 2), (skill 3). I’m interested in (back-end/front-end/full-stack/etc...)”
* If this would be your first job, you can substitute the experience for a project.
* Just be yourself! ☺ Don’t be a robot

**Q: Why do you want to work here?**

* Try to align your skills and goals to the company mission and culture.
* If what the company does excites you, talk about it! Mention any particular team or product you’re aware of.

**Q: Tell me about your favorite side-project.**

* Be sure to mention the impact of this project and why you found it fun! Show passion for whatever you did. You did it in your spare time, after all

**Q: What are you looking for with your next job?**

* Answering this honestly is in the best interest for everyone involved.

**Q: What’s your process when you’re stuck on the problem?**

* If you’re blocked, ideally you should try to solve the problem yourself first. Resources like Stack Overflow and internal company documentation is the best place to start. But, don’t be afraid to ask coworkers if you’re blocked. No one wants you to be blocked for days on end.
* Talk about strategies you use to break down problems, like looking at logs, using a debugger, or even just taking a break.

**Q: Favorite and least favorite language? Why?**

* It would be good to have an answer to this rather than being impartial. It shows experience with multiple tools and what you value as a developer.

**Q: How do you resolve disagreements with coworkers?**

* The way I try to resolve disagreements is as follows:
  + Identify the source of the disagreement – is based on opinion, experience, differences in context?
  + Talk about both sides and come to a conclusion. Often, this can involve exploring both options and involving other engineers if needed. Things like design docs and evaluation criteria can help massively.
  + Try to be open minded, but also don’t be afraid to speak up if you think something can be done in a better way, especially early in your career!

**Q: Tell me about your experience.**

* First, I start by talking briefly about previous jobs, and what I worked on while I was there including which technologies I used.
* Then, if relevant to the company/position, I mention 1-2 projects of mine.

**Q: Talk about a mistake you’ve made.**

* Be honest and show ownership! It’s okay to make mistakes. What they’re looking for is how you deal with them and how you grow after them.
* Be sure to mention what you learned from the occasion and anything positive that came from it down the road (i.e. you improved in a certain area)

**Q: How do you work in fast-paced environments?**

* Most software companies are fast-paced workplaces nowadays. You should be comfortable with ambiguity, iterating on your work, and moving fast in general. That’s just the state of technology at the moment!
* Mention if you have a relevant experience where you had to adjust to ever-changing requirements or deadlines and how you handled it.

No matter what the question is, make sure you be yourself! Be honest. Showing personability is going to be in your best interest. Try to keep your responses short and sweet without rambling much (I definitely have a tendency to do so).

Best of luck ☺