

Assumptions Tripping You Up?

Have you ever worked on developing a product, did what you thought all stakeholders had agreed upon, and then at some point in the development discovered that others had something else in mind? Oops...

You made assumptions and convinced yourself these assumptions were true and treated them as fact as far as the product development was concerned. Don't be too hard on yourself. It happens to most of us. The root cause of most project failures is false assumptions.

Here some suggestions that might help you trip up less often. 😊

1. Learn to differentiate assumptions from facts

When I develop design solutions, I ask myself constantly whether my particular understanding of a requirement and its implementation is based on assumptions or facts. I want to be sure my solution actually meets all requirements – the ones that are written down and the ones that are assumed by stakeholders (spoken or unspoken). Doing this has over time made it easier for me to make that differentiation and it happens automatically and intuitively.

2. Check and re-check

Not sure if you are dealing with an assumption? If you are not sure, assume (yes, I use this on purpose) it is one. Go back to your primary stakeholder(s) and check with them until it becomes crystal clear and thus a fact. Avoid the trap of trying to save time by skipping this step as in the long run it may cost you much more in terms of time and project development costs.

3. Good requirements

One way to avoid having to make a lot of assumptions in the first place is to have really good requirements. I like to write a requirements document if possible and at least create the high-level beginnings of an architecture. The latter can help to flush out my understanding of the design intent against these requirements, hence it provides me with a better grip on facts versus assumptions.

4. Create documentation

Document what you are doing including your intended design solution. Share this with others to ensure they have the same understanding. This is a good way to capture facts and potentially turn assumptions into design requirements and thus facts. Keep documentation lean, but don't skimp. I like to think of it this way: if I need to pick up the design again in 2-3 years, I better have enough documentation to quickly understand what I was doing and why. The latter prevents me from questioning myself all the time.