

just kind of subconsciously “get it”. More screens, really, more anything in your app automatically makes it harder to understand, naturally. Your users won’t understand why there’s two ways to do the same thing, they’ll think that there has to be a difference somehow.

*That makes sense. So, you said concepts. Going back to that to-do example, one of those concepts would be a to-do?*

Right. Like think of the things you can do with a to-do in this theoretical to-do app: you can create one, edit it, delete it, maybe put a reminder on it. That’s a good example actually. It wouldn’t be good to create to-dos and reminders as separate concepts. They should exist in the same system.

*Oh, I totally see what you mean. If you can tell the app to just remind you of a to-do, you don’t need reminders specifically as a separate concept altogether. You need only one way to create anything in your app, a to-do, not two separate ways to create and delete something, right?*

Right, and your user then doesn’t need to learn two ways to do what is essentially the same thing.

*What would you say is it about reminders and to-dos that makes it a good idea to merge them into a single concept? I mean, why do you create the to-do and then put a reminder on it, not the other way around?*

Well, a to-do has everything a reminder needs to be useful, like, some text, a deadline and a checkbox, and a reminder is like an extension of that. In this case maybe you get a notification about your to-do. You can create reminder functionality by notifying a user about a to-do and the other way around it doesn’t really work.

*Why not?*

I mean this is all hypothetical — actually, you’d want to test this kind of stuff with real users to see what they actually intuitively understand. But my intuition is just that the to-do should be the basis of this kind of app, because you can extend a