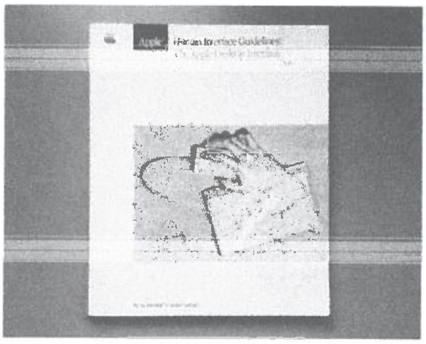
Research: Platform guidelines

Alexis Fogel Follow
Co-founder & CPO @dashlane, passionate about UX/UI, innovation, code, photography and a lot more.
Dec 21, 2016 · 8 min read

Platform guidelines matter and you should follow them

A quick history of guidelines

In 1987, Apple released the Apple 2 computer and with it, the first Human Interface Guidelines. This was a short 162 page book describing how to develop great applications on the Apple platform. The same year, Windows 2 came out with it, guidelines for Windows application development. This is the first time Microsoft started to work with UX designers and a book of guidelines was released in 1993.



Picture of the Apple HIG first book. Courtesy of Bryant Hodson (https://blog.prototypr.io/@bryanthodson)

Both companies understood at that time that in order to drive adoption of their operating systems, good apps were key and that they needed to help app developers if they were to build compliant software. One thing I find astonishing is that many of the good practices we still use today come from these first efforts. Take this example from the first Apple Human Interface guidelines:

> *"To be in charge, the user must be informed. When, for example, the user initiates an operation, immediate feedback confirms that the operation is being carried out, and (eventually) then it's finished… This communication should be brief, direct, and expressed in the user's vocabulary rather than the programmer's"*

Sounds familiar? On the contrary, some have clearly evolved:

> *"People appreciate visual effects, such as animation, that show that a requested action is being carried out. This is why, when a window is closed, it appears to shrink into a folder or icon. Visual Effects can also add entertainment and excitement to programs that might otherwise seem dull. Why shouldn't using a computer be fun?"*

Although animations are still considered to bring value when used appropriately, app developers are warned to avoid using animations that could reduce usability. In the latest Apple HIG, we find:

> *"Avoid gratuitous animation. Animation that serves no purpose or is illogical quickly becomes tiresome and irritating to users. Be sure that the animation you add enhances the user's understanding of your app's functionality."*

Definition of guidelines & major platform guidelines

Ok, so what are guidelines? A simple definition is: *"Recommended practice that allows some discretion or leeway in its interpretation, implementation, or use"*. What I like about this explanation is that it highlights the difference between a true guideline and a rule that you must follow. The distinction is especially important to understand for an iOS app developer, given that Apple is not always crystal clear in its terminology. 'App store review guidelines' for submitting an app are, for you, rules. Ignoring a rule could get your app rejected. But if your app is non-HIG compliant, you will still make it to the app store.

Not rules per se
But recommedations

Right now you have guidelines from three main actors:
1- HIG from Apple for iOS and macOS
2- Material Design guidelines from Android
3- Universal windows platform guidelines

i won't be designing for all platforms

These guidelines are the foundation for a quality experience on each platform. If you take Android for instance, before Google shared its new Material Design standard and guidelines on how to implement it (which, by the way, are truly awesome), the quality of apps on the Play Store was poor, with app developers mainly replicating an iPhone UX on Android

foundation for quality experiences

Today, one could argue that some apps even render better on Android than on iOS. Microsoft recently took a bet with their new guidelines, in creating a way for app developers to use the same code for desktop, mobile and tablets. It is a little early to say whether this will be a success, but I find the approach both bold and interesting, Each of these guidelines takes a unique approach to delivering a consistent user experience, something we all strive for.

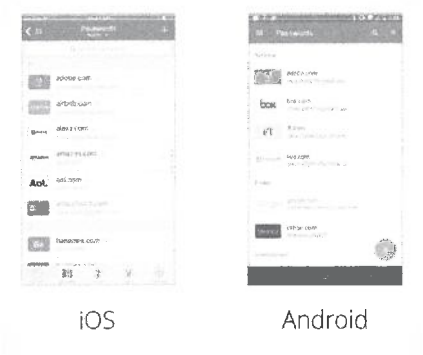
What's in it for you as an app developer?

Usability

Raise your hand if this has never happened to you: You are in a hotel or an Airbnb, you want to take a shower, you get in the bathtub and then it takes you 5 minutes to figure how to switch from the bath to the shower system.
No hands, I bet. And I can tell you that for someone who spends their days working on UX, standing naked in the bathtub wondering how to turn on a shower is a pretty frustrating experience. If only there could be a standardized way to switch to shower mode…
Now the difference between a shower and your app is that I am never going to spend 5 minutes to understand how your app works if I don't need it badly, and so you had better make sure that I get it instantly. That's where guidelines help. Every app from the operating system follows more or less the platform guidelines, which means that people are used to a set of design concepts as they experience them every time they do something on their device. By reusing these same concepts, you ensure that your users will know how to start using your app.
As you can see below, our app on Android and iOS provides the same value but the way to take key actions is totally different, from menu

People will leaves if it's to complicated

interactions to adding new items. And in terms of design, each app feels right for the ecosystem it is a part of.



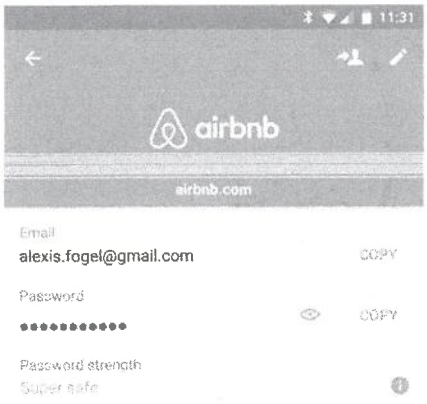
if it walks and feels likeen system app people know how to use it

Design quality and innovation

A very fair question would be: "if I follow the guidelines, how am I going to build anything different from other apps?".
It is true that it is important to stand out from other apps on the app stores. [illegible] individual

First of all, it is not because there are platform guidelines that you don't need a real designer for your apps. Reason number one is that there can be a huge difference between two apps following the same guidelines depending on how these are executed. What's more, do not assume that if you start following guidelines that your app is going to have a great UI/UX right away. What guidelines do provide is a framework for designers to work from. By providing limits through dos and don'ts, guidelines free designers to focus on what is going to differentiate you as a brand. For instance, at Dashlane, we like to provide a personalized experience depending on the services you use. Therefore, as you can see below, when you open an account, we take the main color of the logo (here the Airbnb pink) and use it to customize the whole interface. This is something we do on every platform and that plays well with any guidelines.

following guidelines doesn't mean being bland

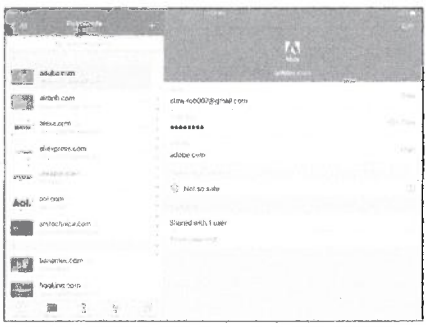


Now of course a simpler answer to differentiation is features. Design and UX are obviously a great way to set yourself apart, but it should never prevent you from looking for key features that will make your app offering stand out from the competition and delight your users (this sounds a bit cliché, but remains true).

Development

So far we have talked primarily about UX and design, but what about implementation? It turns out that most of the time, guidelines are based on standard components that are provided by the platform. This means that by following guidelines when creating your app, you are giving a huge boost to your development team, not only for the first implementation but also for maintenance. Don't take my word for granted and ask developers, I am sure they will tell you that they prefer using standard components rather than re-coding custom behaviors. Here is a good example: Apple standard components are all built for auto-layout, which means that by following them, you app will work beautifully on every device. Before we recreated Dashlane iPhone app to follow Apple guidelines, it took us 2 months to implement the split-view feature on iPad (see below). Now our app works from iPhone 5C to iPad Pro both portrait and landscape without any additional development. And this is even more relevant for Android, considering the diversity of devices on the market.

basic components are provided by the platforms



Team

Almost every company will complain about how difficult it is to find great team members. It is usually considered the most important thing for a startup. Now from my personal experience, the best people I know working on apps are passionate about the platforms they work on and are usually very sensitive about the recommendations from that platform. When I interview engineers, designers and product managers, I always try to get a sense of how well they know the platform guidelines or how willing they are to learn them. It makes for an enthusiastic team, pulling in the same direction.

Distribution & conversion

Another reason why you should consider following guidelines is distribution and other business opportunities. I explained earlier how much platforms care about guidelines, so when they have to choose an app to illustrate innovation, or when the editorial team wants to highlight apps, it is not surprising that they choose those that follow their guidelines. I would never say that the goal of this should be to get featured as it is never a 100% sure thing, but consider that this will always be a plus when talking to partner. To give you a quick example, our relationship with Google really took off when we started working on Material Design with the drive to create an app that fits perfectly into the Android ecosystem. Since then we have been featured a couple of times on the Play Store, we have been awarded "editor's choice" by their editorial team, we have worked on an early program for the fingerprint scanner that was demoed at I/O last year, and we recently announced an open source API in partnership with Google to automate in-app login on Android. Of course, all this did not happen because we decided to follow their guidelines alone, but this clearly helped.

Should you invest in rebuilding your app to follow guidelines?

You are probably in one of the three next situations: either 1) you are already following guidelines and you're now feeling pretty good about yourself, 2) you are going to start a new app and are hopefully considering following platforms guidelines or 3) you have established

The third situation is obviously the toughest. I have been there, and this is not an easy choice to make. Ultimately, the answer depends on your technical and business situation. Is your code dirty enough that it justifies a refactor? Do you plan on building new features that could require reorganizing your app? What does your team think about it? What is sure is that if you are losing time and opportunities because your app is old and out of touch, it might not be a good bet to invest time in a redesign when you have more important things to improve.

I hope this gave a better idea about why platforms guidelines matter. One thing for sure to me is that looking at a platform guidelines will be the first I will do for any new product I create, at least to better understand the environment I am going to evolve in.

i will follow guidelines to avoid 3