

Android vs. iOS Interaction Design Study for a Student Multiplatform App

Abimael Barea, Xavier Ferre, and Lorenzo Villarroel

Universidad Politécnica de Madrid
Campus de Montegancedo
28660 - Boadilla del Monte (Madrid), Spain
{abimaelbarea, villarroel.lorenzo}@gmail.com,
xavier.ferre@upm.es

Abstract. When aiming to develop a multiplatform mobile application or app there is a dichotomy between following each platform interaction philosophy and creating a common cross-platform interaction design. iOS and Android are the two most common mobile platforms used by university students, and they are the obvious choice when planning to develop a mobile app for students in just two platforms. We have compared the particularities of the interaction philosophy of both, and we have studied how four popular apps have addressed the contradiction between following each platform interaction philosophy and having a common interaction design between iOS and Android versions. The results show that there are three possible approaches for multiplatform interaction design, with different pros and cons.

Keywords: Interaction design for mobile platforms, cross-platform consistency.

1 Introduction

User experience has a big effect in mobile application development [1]. Smartphone users enjoy a user experience that is particular to their device mobile platform. According to Wasserman, mobile apps must share common elements of the user interface with other apps in the same platform, and must adhere to platform-dependent user interface guidelines [2]. Mobile application developers need to consider both constraints when creating the interaction design: formal norms in the form of platform-specific user interface guidelines (like [3] and [4]), and 'de facto' norms comprising the common strategies used in the interaction design of the most popular apps. While the former are readily available, it is difficult for developers to get to know the second ones, unless they are highly specialized in a given platform. When an app is designed following both official and 'de facto' norms, users will be able to build on their previous app usage experience, therefore improving their learning curve and their overall satisfaction.

For various market or organizational reasons, most teams must support apps on multiple platforms [1]; therefore maximizing the user base an app may reach. Multiplatform developers hold a dilemma between cross-platform consistency for the

interaction design and offering interaction designs unique to each platform that follow its specific norms. Cross-platform consistency keeps the user experience coherent across multiple platforms [5], but when going against platform conventions it contributes negatively to novice user learnability. Existing generic design guidelines for mobile development, like [6], do not consider these issues, since they address more abstract interaction design concerns to consider when designing for mobility. For the development of a multiplatform app addressed to university students, we have carried out a survey between freshmen, with two aims: First, to choose the two most used mobile platforms as development platforms for the app; and, second, to identify the most popular apps. We have studied popular apps in terms of cross-platform interaction design consistency, to help in the decision of cross-platform consistency vs. platform-specific interaction design.

2 Students Survey about Mobile Platforms and Popular Apps

The survey was delivered to freshmen Computer Science students at UPM, during orientation week in September 2012. 188 responses were received, showing that platform distribution among respondents is as follows: Android (54.25%), iOS (19.68%), BlackBerry (7.98%), Symbian (7.44%), Windows Phone (1.06%) and other (9.57%). Android and iOS cover 74% of the students, so they are the best choice when choosing to develop for just two platforms in this context of use. Given that Android and iOS account together to 85% of the smartphone market [7], this choice may be also adequate for other more generic contexts of use. Participating students were also asked about the three apps they used most. This question received 180 responses, with the following apps mentioned: WhatsApp (67.2%), Twitter (40.56%), Tuenti (30%), Facebook (15.56%), Youtube (6.67%), Safari (5.56%), Gmail (5%), Spotify (3.33%), Shazam (2.22%), and Skype (1.667%). We chose the top three apps (Whatsapp, Twitter, and Facebook) for our study, discarding Tuenti because it is a local Spain-based social network, which would be less representative from a global point of view. Additionally, we considered Skype as well because mobility students typically use this app to communicate with their families, and any app offering information about the campus would consider mobility students as a relevant profile.

3 Comparison of iOS - Android Interaction Design Philosophy

We carried out a study of the iOS and Android versions of the four apps mentioned in the previous section, and the platform stores (iOS App Store and Android Google Play) to understand the main interaction design solutions that users of each platform find in their apps, interpreted according with the advice present in the corresponding platform guidelines ([3] and [4]). It is necessary to look at the interaction design of specific apps in order to fully understand the possibilities offered in platform guidelines. Additionally, we have identified for each app the differences and similarities between iOS and Android versions in terms of interaction design. Table 1 shows the results of the study. The Look column reflects how close the appearance of the user

interface in both platforms is; the Feel column reflects how different the dynamic part of the interaction in both versions is; and the Compliance column reflects the fidelity to each platform user interface guidelines.

Table 1. Comparison of Android and iOS versions of four popular apps

Apps version considered: (iOS; Android)	Look	Feel	Compliance with guidelines	
			iOS	Android
Facebook (v5.0.1; v1.9.10)	The same	The same	Low	High
Skype (v4.1.1310; v2.9.0.315)	Different	Different	Very high	High
Twitter (v5.0.3; v3.4.0)	Similar	Similar	Very high	Very high
WhatsApp (v2.8.4; v2.8.5310)	Different	Different	Very high	Very high

3.1 Main Interaction Design Differences between iOS and Android

After studying the interaction design of these four apps and the platform stores, we have distilled the main interaction design elements that stand out as different between Android and iOS:

- Navigation: Whereas iOS navigation relies on the top-positioned navigation bar, Android offers the action bar on the same position and the (physical or virtual) back button in the bottom part of the screen. iOS navigation bar contains the back button, a title for the current screen and, optionally, buttons in its right part. Android action bar contains the app icon (doubling as the “up” button), a control to alternate views, and action buttons like the search.
- Screen logical regions: Both platforms have a status bar in the top position of the screen, but each one considers different regions in the rest of the screen. iOS divides the rest of the screen in three regions from top to bottom: the navigation bar, the content area, and a tool area that may hold a toolbar, a segmented bar or a tab bar. Android only considers the main action bar on top and the rest is dedicated to a content area, but this content area may be further divided into an optional upper bar, the main content and a lower split action bar to optionally hold additional controls. Android also offers the possibility of a sliding drawer to show a menu hidden under the current view.
- Settings: The overall approach to setting up options strongly differs between both platforms, and it affects to where users expect to find the functionality to change app settings. iOS distinguishes between app settings and app preferences: App settings are accessed through the system settings part of the operating system, and they are expected not to change very often; while app preferences refer to options changed regularly, which are configured from inside the app. Alternatively, Android only considers one kind of setting: app settings located inside the app, and

they must be situated inside the action overflow with the name “Settings”. We have also observed that iOS offers a narrower variety of options and elements for interaction design than Android. Therefore, it is easier to adapt an iOS design to Android than the other way round.

4 Multiplatform Interaction Design Approaches

We have observed three possible approaches to cross-platform interaction design consistency: Platform-Dependent, Own Cross-Platform and Adapted Cross-Platform. A description of each approach, along with their main advantages and disadvantages follows.

4.1 Platform-Dependent

The Platform-dependent approach consists on creating a different interaction design for each platform, closer to each platform interaction philosophy. This is the approach followed by WhatsApp and Skype.

- Pros: The user experience is better adapted to each platform philosophy. There is a higher ease of learning for users, since the delivered experience of use matches their previous experience
- Cons: It is required a higher effort from the development team, since multiple designs need to be created and maintained. Regarding team expertise, more specialized skills in the interaction design for each platform are required.

4.2 Own Cross-Platform

In this approach the interaction design is the same in both platforms, not necessarily following platform norms and conventions. The Facebook app follows this approach.

- Pros: The user experience is coherent between both platforms, so users in one platform may help users in the other one, and users changing from one platform to the other one may directly apply their previous knowledge. There is a lower maintenance effort for the development team, and brand image is reinforced.
- Cons: The app defines its own way of doing things, departing from conventions typically used in other apps. Novel users experience a higher difficulty in learning to use the app, with possible decreased satisfaction.

4.3 Adapted Cross-Platform

This approach is a compromise between the two previous approaches, delivering an app with a similar appearance in both platforms, but with an interaction design that follows platform conventions to a high extent. Twitter follows this approach.

- Pros: The user experience is better adapted to each platform, favoring a positive reaction from the user while still offering some coherence between platforms. With regard to the development team, there is a slightly lower maintenance effort compared to the platform-dependent approach due to both designs sharing the same basic interaction design scheme.
- Cons: The compromise between platforms may leave out some interesting interaction design functionalities present in just one platform. A high level of expertise in each platform interaction design skills is required from (possibly part of) the development team.

5 Discussion

We have carried out a study on the multiplatform interaction design decisions taken in four popular apps in iOS and Android, observing three possible approaches for addressing cross-platform consistency. The choice of a specific approach in a particular multiplatform app development project will depend on business and marketing objectives, available resources and skills in the development team, and usability requirements. When the chosen approach is Platform-Dependent or Adapted Cross-Platform, due to the more constrained variety of options in iOS, it is advisable to conceive first the iOS interaction design than the Android one. Then the resulting iOS-oriented interaction design can be further extended or modified with the increased options offered in the Android platform. We expect to carry out usability studies with users to test how adherence to platform conventions is perceived and the effect it has on app acceptance and user satisfaction.

References

1. Charland, A., Leroux, B.: Mobile application development: web vs. native. *Communications of the ACM* 54(5), 49–53 (2011)
2. Anthony, I.: Wasserman. Software engineering issues for mobile application development. In: *Proc. of the FSE/SDP Workshop on Future of Software Engineering Research (FoSER 2010)*, pp. 397–400. ACM, New York (2010)
3. Apple. iOS Human Interface Guidelines (2012), <http://developer.apple.com/library/ios/documentation/UserExperience/Conceptual/MobileHIG/>
4. Google Inc. and the Open Handset Alliance. Android User Interface Guidelines (2012), <http://developer.android.com/design/>
5. Antila, V., Lui, A.: Challenges in Designing Inter-usable Systems. In: Campos, P., Graham, N., Jorge, J., Nunes, N., Palanque, P., Winckler, M. (eds.) *INTERACT 2011, Part I. LNCS*, vol. 6946, pp. 396–403. Springer, Heidelberg (2011)
6. Hooper, S., Berkman, E.: *Designing Mobile Interfaces*. O’Reilly, Beijing (2011)
7. IDC. Android and iOS Surge to New Smartphone OS Record in Second Quarter (2012), <http://www.idc.com/getdoc.jsp?containerId=prUS23638712>