Evaluating the Mobile Operating System Effectively

Edward Bramanti

November 5, 2013

Abstract

1 Introduction

In today's world, mobile operating systems are now as prevalent as personal computers. More and more, consumers are turning to smartphones and tablets to daily drive their time spent in productivity as well as entertainment. As this field has continued to develop, the consumer is given a myriad of choices between many different operating systems. Hardware and software manufacturers have arisen to provide a competitive environment. This has led to successes for user choice, as hardware and software companies square off to provide the most innovative interface.

While this has been a great success for the consumer, it has also caused problems for firms. A series of patent wars between companies has arisen in the past few years, which has demonstrated the desire for monopoly over the market by utilizing the legal system. Paik and Zhu note that "According to this view, technology firms race to assemble patent portfolios - initially for defensive purposes in the context of a dynamic and competitive field - but then, as the industry matures, convert their shields into weapons to eliminate their competitors in pursuit of market dominance with its platform."[PZ13] This desire to achieve market independence has led many reviewers to create preconceived notions that the market is no longer innovating, but instead creating a "future-proofing" system that fights pointless legal battles to control software innovation.

Because of this new growth in the smartphone market, reviewers are pitting operating system against operating system to help the user discover if it is the best fit for them. The question remains though: Are the journalistic and firm-based reviewers doing an accurate job of evaluating a mobile operating system? In particular, we will be taking a look at Pfeiffer Consulting's "Mobile OS User Experience Shootout" as a reference for how companies conduct evaluations of mobile operating systems today, and how focusing on certain things that connect consumers and developers together through mental process will increase reasoning and understanding behind the results collected from users.

2 Background

To provide background on how evaluations of a mobile operating system should be performed, we will look at the data from the two most important points of view: first, the developer; and then, the consumer, or end-user. Seeing how the developer analyzes the operating system versus the end-user will help to provide a bridge between the mental thought process of the two, and will provide those observing a foundation for how to approach reviewing a mobile operating system effectively.

2.1 The Developer Perspective

Why is the developer's perspective important to evaluating a mobile operating system? Developers are essential to any platform building its ecosystem, and a mobile operating

system is no exception. In addition, understanding the developer's thought process will be invaluable to hearing the other side of the story, since some evaluations of mobile OS' occur only through the eyes of the end user (Pfeiffer's report is an example).

So how do developers evaluate mobile operating systems, and choose which is the best to design for? Palme et al. state that there are six dimensions to how a developer decides on the best mobile operating system for them: "corporate buyer choice, consumer buyer choice, OS vendor's market growth potential, ease of implementation, security, and revenue." [PTSP10] Specifically, ease of implementation is one factor to focus on in determining if a mobile OS is successful. Evaluators generally overlook this, because they focus on the ease of use for consumer instead of the ease of implementation for the developer. Besides, the end-user never sees the developer's thought process, right? This is a common misconception amongst reviewers of mobile OS functionality. The developer's job is to mesh his mental model of how something should work with the user's model. Interaction design itself is defined this way: "A system's designer/developer must effectively communicate his mental model of the system to the system's users through the "image" presented by that system." Therefore, if the "terms and conditions [of the mobile Software Development Kit(SDK)]... limit[s] the freedom of development by restricting the domain of application or variety of functionality," the ease of use for the user suffers. [PTSP10]

It is important to note the importance of the designer in the evaluation of the success of an operating system. Since the ease of implementation is heavily tied to the ease of use for the user, it would be an incomplete analysis of operating system success without considering the developer's perspective on the functionality of the OS at the software development level.

2.2 The End-User Perspective

When many go to evaluate and review mobile operating systems, they run unknowingly into the issue of context. During a controlled evaluation, many users may respond in a certain way to an interface. Some may respond in certain ways when placed in certain situations. But at the end of the day, what is the goal of the end-user? The end-user's experience needs to be evaluated contextually, according to Coutaz et al. The end-user value a contextual experience for their experience of an OS on the go. [CCDG05]

So, why is context a part of the end-user perspective that so many seem to gloss over? It's very difficult to gauge an experience contextually. Apps like Aviate for Android are bringing validity to the importance of contextual computing, and are showing a valid area that is being overlooked by reviewers: what users do at certain moments in certain locations. For example, Aviate considers if you are at the coffee shop by pulling your location from GPS services, and displays a new context on the launch screen with things specifically tailored to your current experience. [Bry] These kind of things matter to the end-user on a mobile operating system. Since the location of a mobile OS is not statically located like desktop operating systems, users will consider context to be essential to the

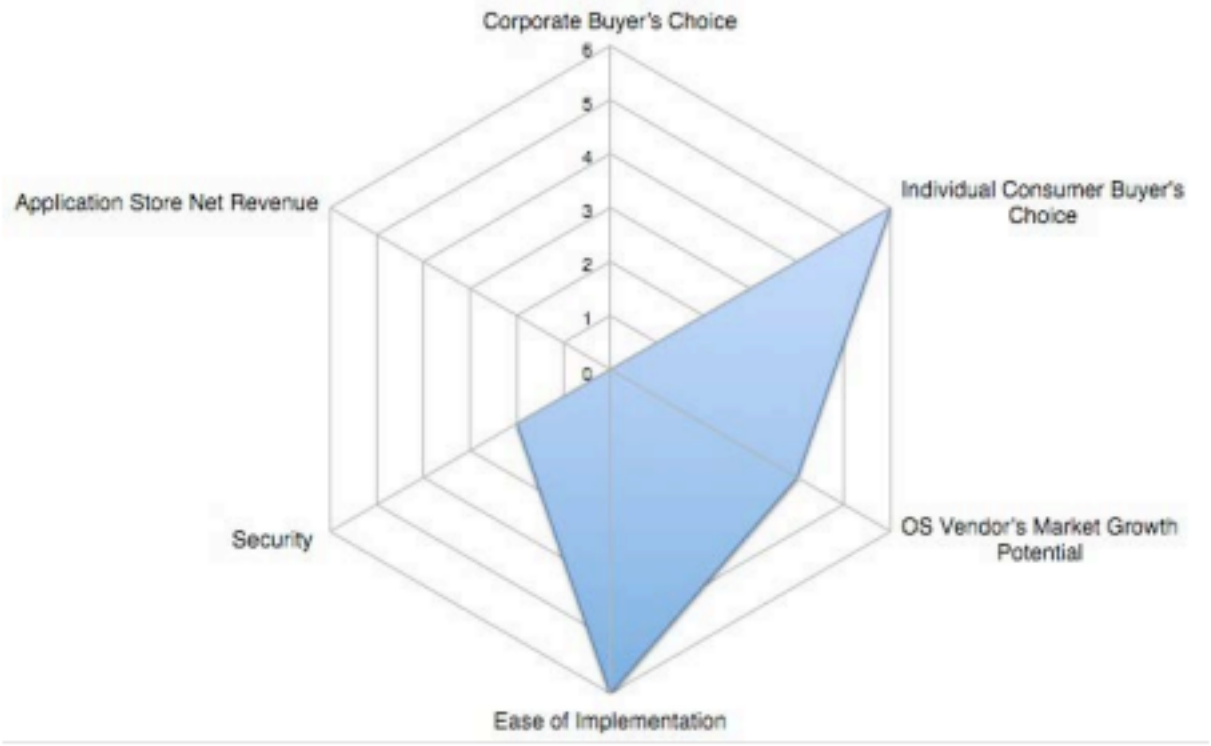

Figure 1: This picture portrays the mental model of the developer choosing a mobile OS to design an app for. [PTSP10]

experience, as can be seen in Figure 2. Evaluations of mobile OS' search for this key term,

3 Methods

4 Discussion

5 Conclusion

References

[Bry] Martin Bryant. Aviate: A gorgeous and smart new android homescreen built for the age of context. http://thenextweb.com/apps/2013/10/15/aviate-a-smart-new-android-homescreen-built-for-the-age-of-context/.

[CCDG05] Joëlle Coutaz, James L. Crowley, Simon Dobson, and David Garlan. Context is key. *Commun. ACM*, 48(3):49–53, March 2005.

[PTSP10] Elia Palme, Chuan-Hoo Tan, Juliana Sutanto, and Chee Wei Phang. Choosing the smart phone operating system for developing mobile applications. In *Proceedings of the 12th International Conference on Electronic Commerce: Roadmap for the Future of Electronic Business*, ICEC '10, pages 146–152, New York, NY, USA, 2010. ACM.

[PZ13] Yongwook Paik and Feng Zhu. The impact of patent wars on firm strategy: Evidence from the global smartphone market. 2013.

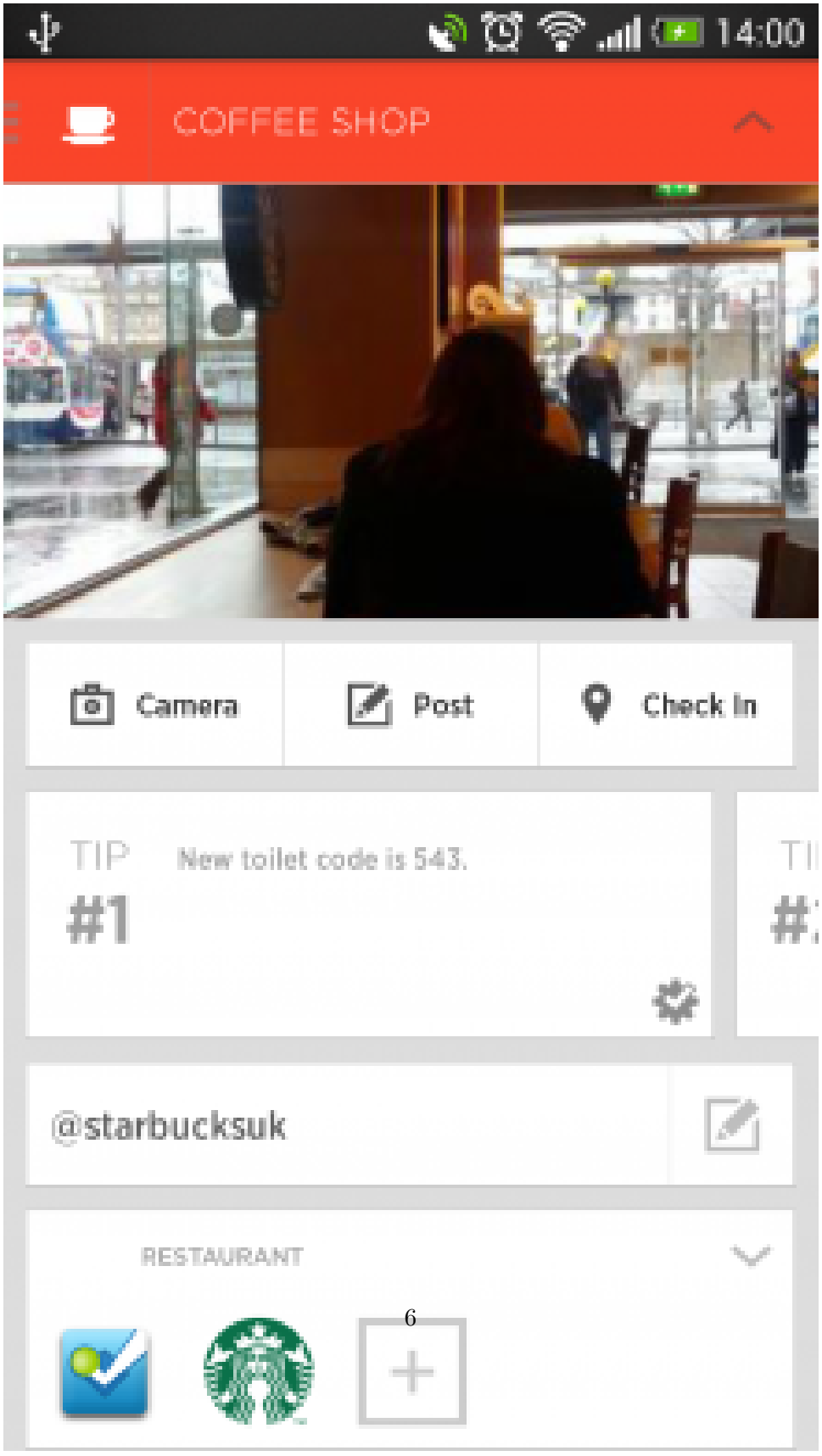

Figure 2: Aviate is a contextual launcher application for Android. Pictured here is the mobile OS detecting the current context of the phone, and modifying its launch screen to show what the user needs in this certain situation. [Bry]