An Exploratory Study of the Adoption of Mobile Development Platforms by Software Engineers

Müller Miranda, Renato Ferreira
Federal University of Pará, Brazil
{mulgsm, renpina}@gmail.com

Cleidson R. B. de Souza
Federal University of Pará &
Vale Institute of Technology, Brazil
cleidson.desouza@acm.org

Fernando Figueira Filho
Federal University of Rio Grande do Norte, Brazil
fernando@dimap.com.br

Leif Singer
University of Victoria, Canada
lsinger@uvic.ca

ABSTRACT

There are several mobile platforms that compete with each other to attract software developers. However, it is not yet well understood which factors developers take into account when deciding on a particular platform. We report on an exploratory study that aims to address this gap. Through semi-structured interviews that used diffusion of innovations theory as conceptual framework, we identified some of these factors. For instance, we uncovered that developers perceive the Android platform as more accessible and compatible with their existing knowledge, but that they fear its fragmentation. Some developers choose iOS simply because sales are more lucrative on that platform. Our preliminary findings can help developers to decide which platforms to use and platform vendors to optimize their offerings to developers.

Categories and Subject Descriptors

H.1.2. [**User/Machine Systems**]: Human factors.

General Terms

Human Factors.

Keywords

Diffusion of Innovations, Software Ecosystems, Adoption, Platforms, Mobile, Android, iOS.

1. INTRODUCTION

According to Bosch and Petra [1], a "software ecosystem" consists of a software platform, a set of internal and external developers and a community of domain experts in service to a community of users that compose relevant solution elements to satisfy their needs. Facebook, Windows Kinect, SAP, and Linux are examples of software ecosystems for different domains.

There are several mobile ecosystems such as Android, iOS, Windows Phone, or Firefox OS. Butle [6] discusses the evolution

of mobile platform market share from 2009 to 2010 for various mobile platforms and shows that some mobile ecosystems are adopted more than others. More recently, Vision Mobile [7] compared the popularity of mobile platform in different regions (cf. Fig. 1).
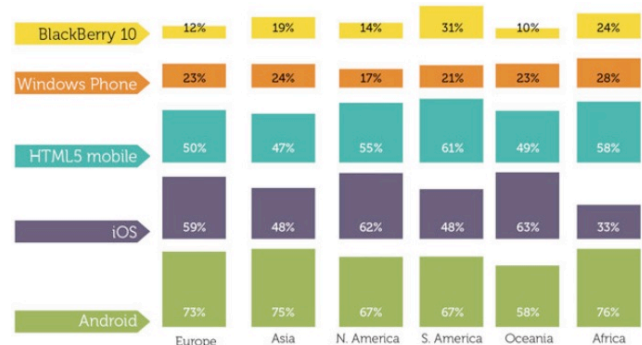

Figure 1: Percentage of developers using each platform, by Region (2013).

The availability of applications for a platform directly influences consumers' buying decisions, so attracting software developers to a mobile ecosystem can make a crucial difference for companies competing for market share. Several factors can influence the perception of a platform's qualities: the programming language used, the qualities of the software development kit (SDK) available for a platform, cost, hardware requirements, or the customer base a developer could reach when deciding on a platform. In short, identifying the factors that software developers take into account when deciding on a mobile development platform is an important research question. This is exactly the question that we are addressing in our joint research project. This paper presents initial results from our first exploratory study. Using Roger's diffusion of innovations theory [4] as our conceptual framework, we concentrated on five attributes of mobile development platforms: relative advantage, compatibility, complexity, observability, and testability. According to Rogers [4], these attributes are crucial determinants of an innovation's adoption rate. An innovation in this context then refers to a mobile development platform.

Through semi-structured interviews with iOS and Android developers as well as a subsequent qualitative analysis, our study aims to understand which factors are important to developers when adopting a mobile ecosystem. The interviews and analysis we conducted were guided by Rogers' innovation attributes. We

mapped these attributes to socio-technical aspects that can be inquired about and evaluated. For example, the *complexity* attribute can be mapped to the (non-)existence of tools and example applications.

2. SOFTWARE ECOSYSTEMS

The last decades have witnessed several strategies for managing the complexity of large-scale software development, including Software Product Lines (SPL) and Software Ecosystems [1]. SPLs typically consist of a software platform shared by a set of products. Each product can typically select and configure components in the platform for its own purposes and extend the platform with product specific functionality. Software ecosystems extend the SPL strategy by involving a much broader set of actors, such as service and infrastructure providers, users and external developers. Instead of relying solely on intra-organizational interactions, the software ecosystem approach takes a "community perspective", and hence actors require a community-centric way of collaborating and coordinating [1]. In fact, the success of a software ecosystem largely depends on how effective the "ecosystem" and its constituent parts are at attracting external actors to the platform.

In this paper, we analyze mobile software ecosystems in three different dimensions: technical, business and social. The technical dimension involves the software platform and infrastructure supporting a particular ecosystem. The business dimension involves business models, licensing and sales strategies, etc. The social dimension defines how actors interact to achieve their goals [5]. This paper addresses the perspective of software developers regarding each of those three dimensions and uses the Diffusion of Innovations Theory to contextualize our preliminary findings.

3. DIFFUSION OF INNOVATIONS THEORY

The Diffusion of Innovation theory developed by Everett Rogers, seeks to explain the process of diffusion of an innovation, as well as why an innovation is accepted or rejected by individuals or units adopters [4]. In the context of this work, the intention is to examine the factors related to acceptance or rejection of mobile platforms by software engineers.

According to Rogers, innovation is defined by ideas, practices or objects that are perceived as new by the individual to others even if somewhere in this "innovation" is already outdated, i.e., the perception of the idea of newness, for the individual determines his reaction to it. If an idea seems new to the individual, is an innovation.

The characteristics of an innovation help us explain their different rates of adoption: according to Rogers, a variation from 49% to 87% in the rate of adoption can be explained by five attributes, namely (i) relative advantage, (ii) compatibility, (iii) complexity, (iv) testability and (v) observability.

The *Relative Advantage* refers to how improved an innovation is perceived to be over the previous generation. Better tools and reduced costs for development are typically considered advantages when deciding to adopt an innovation. *Compatibility* is the degree to which an innovation is perceived as consistent with the individual's daily activities, past experiences and needs. *Complexity* is defined as the degree to which an innovation is perceived as complicated or difficult to use, so the simpler an innovation is perceived, the greater is the likelihood of it being adopted. *Testability* is defined as the degree to which the

innovation can be tried before being adopted. Finally, *Observability* is the degree to which the benefits of an innovation are visible to others. These five attributes were used in the data analysis, as we describe in the next section.

4. STUDY DESIGN

This study is the first of a series of studies we want to perform and that are part of our research project. Due to the exploratory nature of our investigation, a qualitative approach was chosen.

We conducted semi-structured interviews with nine mobile developers (eight men and one woman), of these 5 were actively developing for Android, 3 for iOS, and 1 developed for both platforms. Our interviewees aged between 19 to 29 years, and they were all from Brazil. Two were students (one undergraduate and one graduate) with less than 2 years of experience developing mobile apps. The others were professionals with over two years of experience in mobile development. The interviews were conducted in the period of August to December of 2013, through a non-probability sample.

Upon completion of data collection, interviews were analyzed using grounded theory [8] techniques. More specifically, the open axial and coding was performed resulting in 265 categories identified. Later on, these categories were mapped back to Rogers' innovation attributes.

5. PRELIMINARY FINDINGS

We present our preliminary findings in the following subsections according to the five innovation attributes from Rogers' Diffusion of Innovation Theory introduced earlier in this paper.

5.1 Testability

The first finding of this work suggests that the Android platform is presented with a greater degree of testability, when compared to iOS. The variety of models and development operating systems were among the reasons for this greater testability. In particular, a developer is not required to have a specific computer for developing and testing applications, which is often the case for the iOS platform. One interviewee mentioned: "*Android is the operating system that you are looking in the world, [it] should have more than 50% of people using it, because Android has several models, ... [it is] affordable.*"

Despite Apple's and Android's software development kits being provided free of charge, there are some important differences in the distribution of applications. Android developers typically have another ways to build, test and distribute their applications on real devices without any charge: the developer just need to generate an executable application and can use various channels for distribution, e.g., email, flash drive, DVD etc.

One respondent stated that to publish in Google Play, the developer needs to pay a one time fee of USD $ 25. In that regard, the iOS platform does not offer the same flexibility: the developer willing to distribute their application would typically need to create an account and purchase a development license ranging from $99 to $299 annually, depending on the type of account. Moreover, the app can only be installed for testing in a limited number of devices (100). One informant provided this information in his interview: "*To test apps on your iOS device, you need two things: an Apple machine, due to the operating system and a developer account to run your application on your phone, it's $99 a year.*"

5.2 Observability

Our results suggest that the Android platform has the highest degree of observability compared to iOS by providing a larger audience of developers. One of our informants, said: "*It is very difficult you find someone who has an iPhone or Mac Book. Android ... now you just have Eclipse and any operating system to develop, and more, the amount of people that have some Android smartphone is larger than those with the iPhone.*"

5.3 Compatibility

In general, Java as a programming language is more widespread than Objective-C both at universities and private companies. Therefore, it is possible to identify a greater compatibility with Android by using the Java language than the iOS platform that uses Objective-C. One of our informants said: "*I found it very similar to some frameworks in Java in which I've had worked, then I liked and I chose to continue developing for Android*".

In addition, two other factors have been identified: the Eclipse IDE for Android development is more compatible than the Xcode IDE for iOS. The mobile operating system, in this case Android, is independent and runs on Windows, Linux and Mac OS X. On the other hand, development for the iOS platform is dependent on the Mac OS X operating system.

5.4 Complexity

Although our results identified a higher degree of compatibility for the Android platform, our informants informed that the complexity of developing applications for the iOS platform is smaller than for Android *after the first steps to get some basic working knowledge*. As reported by one of our informants: "*Objective-C is very different from any object-oriented language, ... method calls, methods for object creation are completely different from object-oriented [languages] we are used to, such as Java. But from the moment that you break that barrier, it becomes much simpler to develop than Android.*"

In addition, one can note that the Android platform is present in handsets from different brands and models (Samsung, Sony, Motorola, LG etc). Each manufacturer modifies the system with the intent to customize it to their devices. A developer, says, "*it was more interesting to develop for iOS, because you worry less about various layouts.*" So, there is greater difficulty in adapting applications to different types and screen sizes as well as to some hardware limitations. Because the iOS platform is restricted to Apple's products, there is less variation in size and hardware configurations. This reduces the time developers spend focusing on screen layouts, different configurations, and so on.

Another interesting point revealed in our interviews is that emulators are very limited when compared to real mobile devices. This problem has been cited by many of our respondents. It is not always possible to test all the features of real devices in the emulator. For example, temperature sensors, accelerometers, gestures, etc. This is exemplified by one of our informants that "*Not everything you can simulate on the emulators, then it is a bit complicated to develop only using the emulator. The best is to test with the mobile device*". Also, another factor that has a great impact during the tests is that the emulator has much lower performance compared to a real device. And this is a problem observed on both platforms. Asked if this is reflective only for computers with weak hardware configurations, he replies, "*Only if the computer is very poor. I had the experience of even switching to a better computer, [but] I got the same result.*"

5.5 Relative Advantage

This feature is subjective to each developer. A developer who chooses the Android platform for creating applications may have taken into account their great popularity, or financial ease in order to select this innovation, for example. For some, the fact that the Android platform is more popular is important, as a developer says: "*Many people use Android, the majority of the population, then I would develop for a platform which I know enough people would use it.*" A user points out a financial advantage, because in the "*AppStore you get a much better return than on Google Play, I saw this with the experience of two colleagues. They launched a very similar application, a game about Naruto and you could see that in the App Store it sold ten, twenty times more than in the Android Market.*"

6. DISCUSSION

The work described in this paper aims to investigate the relationship between the supplier of the central platform (e.g., Google or Apple) with the software engineers who develop apps for that platform. According to our results, we can conclude that the Android platform provides a more flexible development environment because it does not depend on specific hardware and software, thus lowering the barriers for developers. Our interviewees also reported a higher degree of testability, observability, and compatibility and complexity for Android in comparison to iOS (see section 5).

These results reflect very well Goadrich's and Rogers' [11] work who present a comparative analysis of development between the Android and iOS mobile platforms. These authors aim to identify what platform the academia should teach undergraduates. In this comparison the authors take into account the requirements of both software and hardware platforms, e.g., graphical user interface, documentation, Software Development Kit, etc. In conclusion the authors identified that the iOS compared to Android development can be more difficult because of the need for specific hardware and tools, and suggest that language Objective-C and Xcode IDE programming for iOS is less common than the combination of Eclipse and Java.

On the other hand, it is important to note that this research also points out that the Android platform is presented with a greater degree of complexity when compared to iOS. This is based on the answers from the informants who find iOS development easier *after* getting more experienced with it. They also reported being able to develop applications with a fewer lines of code, because the platform offers features that simplify the development of applications. To the best of our knowledge, this is a novel result from our research.

Another result of this work refers to the limitation of emulators for not being possible to test all functionality related to hardware, for example, the temperature sensors. There are difficulties of such kind in both platforms. Joorabchi and colleagues [12] seek to obtain an understanding of the key practices and challenges faced by developers for developing applications for different mobile devices on native platforms, among them also point out the need for better emulators.

Some authors [13] [14] seek to understand what motivates a person to develop for a specific mobile platform. Pappas [13], for instance, shows that the iOS platform is chosen by developers who are interested in income. This reinforces the results presented in Section 5.5 when we discuss the relative advantage of mobile platforms.

Begel and Sillito [14] show social characteristics for developing for the Windows Phone, as a potential upgrade in the developers' curriculum, hobby or just the fact these developers remain relevant to the labor market in the context of mobile software ecosystems. We did not observe such motivations in our interviewees, but plan to investigate these aspects in future work.

7. CONCLUSION AND FUTURE WORK

This study attempted to better understand the scenario of application development for mobile ecosystems based on the developers' perspective. We used semi-structured interviews based on Rogers' diffusion of innovation theory, while the analysis was conducted using grounded theory techniques. With the results of the interviews, it was possible to gather different types of insights. For example, the reasons why the Android platform is regarded as more popular than the iOS platform or how the experience of the adopter with a particular technology may influence his/her perception of the innovation that arises as a result, especially if they relate to similar technologies or ideas.

We recognize that the results of our study are preliminary, but we believe that they provide interesting insights about the software developers' experience with mobile platforms. Of course, there are several avenues for improvement, and consequently, future work. First of all, it is possible to enhance the interview guide to ask new questions and improve our understanding of app development. For instance, at no time questions about backward compatibility between applications were asked. In addition, several other points should be addressed or further developed, such as the platform which has the most active and participatory community and how this contributes to their adoption, strengths and weaknesses of the documentation available in each ecosystem, different ways to emulate, etc. We plan to gather this additional information through new interviews.

8. ACKNOWLEDGMENTS

9. REFERENCES

[1] Bosch, J. and Petra, M. "Software product lines, global development and ecosystems: Collaboration in software engineering", In: Collaborative Software Engineering, Edited by Ivan Mistrk, Andre van der Hoek, John Grundy, and Jim Whitehead & Springer Berlin Heidelberg (2010).

[2] Cusumano, M. "The Business of Software: What Every Manager, Programmer and Entrepreneur Must Know to Succeed in Good Times and Bad." Free Press, New York, (2004).

[3] Jansen, S. and Cusumano, M. "Defining Software Ecosystems: A Survey of Software Platforms and Business Network Governance.", In: Proceedings of the international Workshop on Software Ecosystems, (2012).

[4] Rogers, E. "Diffusion of innovations". Free Press, New York. (1995)

[5] Barbosa, O., Santos, R., Alves C., Werner, C., Jansen, S. "A Systematic Mapping Study on Software Ecosystems through a Three-dimensional Perspective". In: Software Ecosystems: Analyzing and Managing Business Networks in Software Industry. Edward Elgar, Cheltenham, UK. (2013).

[6] Margaret Butle "Android: Changing the Mobile Landsscape ", In: Pervasive Computing, IEEE (2011).

[7] Vision Mobile. "Developer Economics Q3 2013: State of the Developer Nation".

[8] Strauss, A. and Corbin, J. "Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory". Sage Publications, (1998)

[9] Seaman, C. "Qualitative Methods". In: Guide to Advanced Empirical Software Engineering. Springer London, (2008).

[10] Charmaz, K. "Constructing Grounded Theory. A Practical Guide Through Qualitative Analysis". Sage, London. (2006).

[11] Mark H. Goadrich, Michael P. Rogers. "Smart smartphone development: iOS versus android". In: Proceedings of the 42nd ACM technical symposium on Computer science education, (2011).

[12] Mona Erfani Joorabchi, Ali Mesbah, and Philippe Kruchten. "Real challenges in mobile app development". In: Proceedings of the ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, (2013).

[13] Andreas Pappas "How do developers prioritize platforms? iOS vs Android vs HTML5" Vision Mobile (2013). Retrieved January 2nd, 2014, from www.visionmobile.com/blog/2013/12/developers-prioritise-platforms-ios-vs-android-vs-html5/

[14] Sillito, J., & Begel, A. "App-directed learning: An exploratory study." In: The 6th International Workshop on Cooperative and Human Aspects of Software Engineering, San Francisco, CA, USA (2013).