

Relink-MobileAppData: Master Project Specification

Dr. Sebastiano Panichella

1 CONTEXT

In the context of software evolution, researchers conceived and experimented a wide spectrum of approaches to extract pertinent information from software repositories with the aim of supporting software developers during maintenance tasks. In particular, software evolution research has focused on the analysis of structured and semi-structured artifacts (e.g. source code and versioning data) of traditional software systems like the Firefox web browser or software projects coming from open source communities such as the Apache Software Foundation (ASF). However, very few research studies were performed for supporting mobile app development process. Emerging domains, such as mobile devices, are growing rapidly and the mobile app development industry has expanded and now it is growing year after year. Indeed, market studies suggest that the global mobile app economy is expected to be worth \$143 billion by the end of 2016 [1]. Recent studies indicate that apps receiving high users' ratings use APIs that are less fault- and change-prone than the APIs used by low rated apps [2]. Thus, in the context of mobile software development, developers must pay attention to building robust and reliable apps. In fact, users easily get frustrated by repeated failures, crashes, and other bugs; hence, they abandon some apps in favor of their competitors.

2 MOTIVATION

Mobile apps are designed, implemented, tested and distributed in a different manner than traditional software [3], [4], [5] with the aim at efficiently answering the mobile market requests. Beside that, also the distribution mechanism for mobile apps is very different to the distribution of traditional software products. Mobile apps are released through the app markets rely on app stores, such as Google Play and Apple's App store. Developers of both large (Facebook, Youtube, Adobe etc.) and small companies (having a number of developers < 6) release their apps providing the same mechanism for the users to download and install them. Thus, the data stored in app stores represents an interesting source of information for software engineering researchers, since they *"have never had available, a more rich, wide and varied source of information about software products"* [3], [4]. The data that these mobile app markets contain can be used to generate new research and perform further empirical results with the aim to support developers during mobile app development.

3 RECENT WORK

Mobile apps are designed, implemented, tested and distributed in a different manner than traditional software [3], [4], [5] with the aim at efficiently answering the mobile market requests.

Differently from traditional software, the app markets allow users to post reviews of mobile apps. Thus, mobile software developers get continuous feedback from users that can be

leveraged to help them improve their applications. Prior work suggest that such information can be used to leveraged user reviews to extract user-faced issues [6], and new requirements [7]. However, one of the main challenges of mining data of mobile app is that app developers spend a considerable amount of time in collecting and exploiting user feedback to improve user satisfaction for three main reasons: (i) approximately only one third of the information contained in user reviews is helpful for developers [8]; (ii) app stores include a *substantial body* of reviews, for example, mobile apps receive approximately 23 reviews per day (popular apps, such as Facebook, received on average 4,275 reviews per day) [8], which requires a large amount of effort to manually analyze and process; (iii) the *quality* of reviews varies greatly, from useful reviews providing ideas for improvement or describing specific issues to generic praises and complaints [8].

In the last years researchers started to focus more in depth the software engineering issues related to the context of mobile software [9]. First of all, recent work focused on issues related to testing mobile apps [13], code reuse in mobile apps [14], device fragmentation [6] and mobility issues in mobile environment [15]. Other work instead focused on resource usage and optimization [12], monetization of apps [11] and problems related to mobile security [10].

4 GOAL OF THIS MASTER PROJECT

In a recent in-depth study of user feedback for mobile app, Sebastiano Panichella *et al.* proposed a first *feedback-mechanism* to help developers to classify user reviews according to specific development tasks: fixing bugs, implementing new features and/or improving existing feature [16]. This work revealed that Natural Language Processing (NLP), Textual Analysis and Sentimental Analysis can be exploited to automatically identify and classify useful feedback in according to specific maintenance tasks. The tool implementing this research approach is already available for download ¹.

Stemming from the preliminary results on automatic user reviews classification, we propose to investigate further methods, tools, and techniques able to mine the sheer amount and diversity of the potentially available data, combine them to provide to developers the appropriate feedback-environment they need to perform the development and maintenance tasks performed for the evolution of mobile applications. Therefore, the scientific challenge is to modelling the information presented to app developers to augment the knowledge at their disposal and to support mobile app evolution.

5 TASK DESCRIPTION

The main tasks of this projects are:

1. <http://www.ifi.uzh.ch/seal/people/panichella/tools/ARdoc.html>

- 1) *Traceability*: design and implement a tool able to
 - Linking requirements from reviews to source code;
 - Linking reviews to specific versions;
 - Linking app store metadata (price, rating, etc.) to source code;
 - Update and improve requirements using information contained in reviews (this task is OPTIONAL since it is very challenging).
- 2) Predict the right price of an application (this task is OPTIONAL, however, if the students involved in the project are interested they can replace one of the *traceability tasks* described above with this task);
- 3) Conceive a mechanism which uses reviews data to check feeling of users about new releases (this task is OPTIONAL, however, if the students involved in the project are interested they can replace one of the *traceability tasks* described above with this task);
- 4) All the previous mechanisms should be able to work on both Google Play and Apple's App store data.
- 5) *Evaluation*: the usefulness of the implemented tool will be evaluated by performing a study involving professional developers. We have established connections with developers and testers of *Sony Mobile*². Thus, we plan to perform such experiment involving developers of Sony Mobile in their working context. Students working in the master project are also encouraged to invite other developers of companies from their contact list.

The minimum number of students that should participate to the project is 2. A number of 3 – 4 students would be ideal for this project.

Dr. Sebastiano Panichella will advice and monitor the research activity of the students involved in this project to achieve the goals of the project. Thus, weekly meeting will be scheduled with the students involved in the project in order to monitor the progress and status of the project.

REFERENCES

- [1] A. Pappas. *Developer economics: App market forecasts.* , 2013-2016, <http://www.visionmobile.com/blog/2013/07/developer-economics-app-market-forecasts-2013-2016/>
- [2] M. Linares-Vásquez, G. Bavota, C. Bernal-Cárdenas, M. Di Penta, R. Oliveto, D. Poshyvanyk *API Change and Fault Proneness: A Threat to the Success of Android Apps.* , In Proceedings of the 2013 9th Joint Meeting on Foundations of Software Engineering (FSE 2013), ACM, Saint Petersburg, Russia, 477-487.
- [3] A. Al-Subaihini, A. Finkelstein, M. Harman, Y. Jia, W. Martin, F. Sarro, Y. Zhang, *App Store Mining and Analysis.* , In Proceedings of the 3rd International Workshop on Software Development Lifecycle for Mobile, (DeMobile 2015), ACM, Bergamo, Italy, 1-2.
- [4] M. Harman, J. Yue Jia, Z. Yuanyuan, *App store mining and analysis: MSR for app stores.* , In Proceedings of the 9th IEEE Working Conference on Mining Software Repositories (MSR 2012), 108-111.
- [5] N. Viennot, E. Garcia, J. Nieh, *A Measurement Study of Google Play.* , In Proceedings of the 2014 ACM International Conference on Measurement and Modeling of Computer Systems, 2014, Austin, Texas, USA, 221-233.
- [6] H. Khalid, M. Nagappan, E. Shihab, A. Hassan, *Prioritizing the Devices to Test Your App on: A Case Study of Android Game Apps.* , In Proceedings of the 22Nd ACM SIGSOFT International Symposium on Foundations of Software Engineering, (FSE 2014), ACM, Hong Kong, China, 620-620.
- [7] C. Galvis, V. Laura, K. Winbladh, *Analysis of User Comments: An Approach for Software Requirements Evolution.* , In Proceedings of the 2013 International Conference on Software Engineering, (ICSE '13), IEEE, San Francisco, CA, USA, 582-591.
- [8] D. Pagano, W. Maalej. *User feedback in the appstore: An empirical study.* , In Proceedings of 21st IEEE International Requirements Engineering Conference (RE), 2013 , 125-134.
- [9] E. Shihab, Y. Kamei, P. Bhattacharya, *Mining challenge 2012: The Android platform.* , In Proceedings of the 9th IEEE Working Conference on Mining Software Repositories (MSR 2012), 112-115.
- [10] P. Riedl, P. Koller, R. Mayrhofer, A. Möller, M. Koelle, M. Kranz, *Visualizations and Switching Mechanisms for Security Zones.* , In Proceedings of International Conference on Advances in Mobile Computing (MoMM '13), ACM, Vienna, Austria, 278-281.
- [11] J. Gui, S. Mcilroy, M. Nagappan, W. Halfond, *Truth in Advertising: The Hidden Cost of Mobile Ads for Software Developers.* , In Proceedings of the 37th International Conference on Software Engineering (ICSE '15), IEEE, Florence, Italy, 100-110.
- [12] K. Aggarwal, A. Hindle, E. Stroulia, *GreenAdvisor: A Tool for Analyzing the Impact of Software Evolution on Energy Consumption.* , In Proceedings of the International Conference on Software Maintenance and Evolution, (ICSME 2015).
- [13] A. Machiry, R. Tahiliani, M. Naik, *Dynodroid: An Input Generation System for Android Apps.* , In Proceedings of the 9th Joint Meeting on Foundations of Software Engineering (ESEC/FSE 2013), ACM, Saint Petersburg, Russia, 224-234.
- [14] I. Ruiz, M. Nagappan, B. Adams, E. Hassan, *Understanding reuse in the Android Market.* , In Proceedings of the 20th International Conference on Program Comprehension (ICPC), 113-122.
- [15] J. Zhang, A. Hämäläinen, J. Porras, *Addressing Mobility Issues in Mobile Environment.* , In Proceedings of the 1st Workshop on Mobile Middleware: Embracing the Personal Communication Device (MobMid '08), ACM, Leuven, Belgium, 3-5.
- [16] S. Panichella, A. Di Sorbo, E. Guzman, C. Visaggio, G. Canfora, H. Gall. *How Can I Improve My App? Classifying User Reviews for Software Maintenance and Evolution.* , In Proceedings of the International Conference on Software Maintenance and Evolution, (ICSME 2015).

2. <http://www.sonymobile.com/>