

Integrating Adaptive Mechanisms into Mobile Applications Exploiting User Feedback

Quim Motger¹, Xavier Franch¹, and Jordi Marco²

¹ Department of Service and Information System Engineering (ESSI), Universitat Politècnica de Catalunya (UPC), Jordi Girona 1-3, 08030, Barcelona, Spain
{jmotger, franch}@essi.upc.edu

² Department of Computer Science (CS), Universitat Politècnica de Catalunya (UPC), Jordi Girona 1-3, 08030, Barcelona, Spain jmarco@cs.upc.edu

Abstract. Mobile applications have become a commodity in multiple daily scenarios. Their increasing complexity has led mobile software ecosystems to become heterogeneous in terms of hardware specifications, features and context of use, among others. For their users, fully exploiting their potential has become challenging. While enacting software systems with adaptation mechanisms has proven to ease this burden from users, mobile devices present specific challenges related to privacy and security concerns. Nevertheless, rather than being a limitation, users can play a proactive role in the adaptation loop by providing valuable feedback for runtime adaptation. To this end, we propose the use of chatbots to interact with users through a human-like smart conversational process. We depict a work-in-progress proposal of an end-to-end framework to integrate semi-automatic adaptation mechanisms for mobile applications. These mechanisms include the integration of both implicit and explicit user feedback for autonomous user categorization and execution of enactment action plans. We illustrate the applicability of such techniques through a set of scenarios from the Mozilla mobile applications suite. We envisage that our proposal will improve user experience by bridging the gap between users' needs and the capabilities of their mobile devices through an intuitive and minimally invasive conversational mechanism.

Keywords: mobile software ecosystems · mobile applications · adaptive software systems · chatbots · conversational agents · user feedback · user categorization

1 Introduction

The pervasiveness of smartphone devices and the use of mobile applications as fundamental tools around the globe is already a reality. By January 2020, around 5,190 million unique users were using a mobile phone [9], which implies an increase of 2.4% compared to January 2019 reports. Despite being far from universal, and even though there are still significant usage gaps (e.g., rural-urban, gender or age), a wide range of services including instant messaging and social networking have become essential in a variety of personal and business cases [1].

This increasing trend has a direct impact not only in the number of users, but also in the complexity and heterogeneity of mobile software ecosystems. This complexity entails hardware specifications (e.g., manufacturers, sensors), software specifications (e.g., operating systems, mobile applications), features (e.g., web browsing, social networking) and context of use (e.g., personal vs. professional, private vs. public), among others. As users' goals and needs are unique, managing and fully exploiting the potential of the applications portfolio is a challenging task, which includes not only the ability of such ecosystems to unveil and adapt hidden features or settings from the user viewpoint (e.g., detecting and understanding hidden privacy settings [5]), but also their ability to adapt hardware and software components to match the users' needs.

A suitable method to overcome these challenges is the integration of adaptation mechanisms and decision-making features in software systems [18], a research area undergoing intense study in mobile-based systems [7]. However, smartphones present additional challenges related to the management of personal and sensitive data, as well as the additional knowledge that can be extracted from this data [15]. Therefore, they require from explicit user interaction for most changes in their mobile devices and applications.

Despite this limitation, the need for integrating the user in the adaptation loop can be conceived as an opportunity. In fact, effective evaluation of explicit user feedback is considered a key factor and a challenge in adaptive software systems [8]. Among explicit feedback collection techniques from users, the integration of virtual assistants, conversational agents or simply *chatbots* is beginning to grow exponentially. Nevertheless, chatbots are still an emerging technology, with a lack of scientific relevant presence in some research areas [2].

In this paper, we aim to depict a work-in-progress proposal of an end-to-end framework for **semi-automatic adaptation in the context of mobile software ecosystems**. We refer to semi-automatic adaptation as the set of mechanisms which combine (1) autonomous techniques for the collection of contextual and implicit feedback data, and (2) explicit feedback knowledge obtained through users interaction with a chatbot integrated in their smartphones.

2 Background

From the software engineering viewpoint, mobile-based systems or **mobile software ecosystems** as defined by Grua et al. [7] can induce adaptation mechanisms anywhere in the system. It is relevant to differentiate between adaptation at device level (e.g., mobile platform, device hardware) or at mobile application level (e.g., client app, back-end). With independence of where the adaptation takes place, one of the underlying goals of these approaches is to improve user satisfaction and the Quality of Service (QoS) delivered to these users [7].

If we focus on adaptation at the **mobile application** level, mechanisms and strategies to achieve these goals are diverse. Some examples include dynamic selection of user interfaces to improve usability dimensions at runtime [3], context-aware self-adaptation techniques for back-end infrastructures of mobile

applications [14], and automatic reconfiguration of features to achieve customizable and adaptive personal goals [16]. On the other hand, the level of **automation** of these mechanisms is also manifold, whether adaptation is designed as a fully-automatic process transparent to users or as an assisted, semi-automatic process requiring explicit interaction between users and the application itself [4].

In practice, most of these adaptation strategies and mechanisms are transparent to the users. When measuring and evaluating adaptive mobile applications, evaluation focuses on QoS to enhance user experience [17]. Among these quality characteristics we highlight functional or **feature** suitability based on users needs, and non-functional or **quality** concerns including usability, reliability and performance efficiency as the most cited characteristics in mobile apps [10].

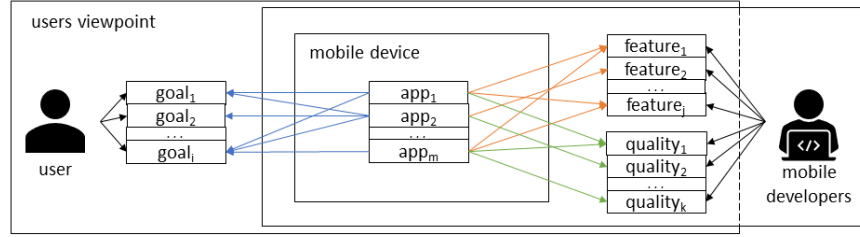


Fig. 1. Mobile software ecosystems from the user viewpoint

All things considered, from the **users' viewpoint**, mobile software ecosystems are conceived as support tools giving them access to a portfolio of **mobile applications** exposing a set of **features** and **qualities** to achieve their personal **goals**. In the context of enacting these ecosystems with adaptive tools, from the **mobile developer viewpoint**, the challenge is to identify, define, develop and integrate the underlying adaptation mechanisms these features and qualities from mobile applications require to help users achieve these goals.

To this end, context monitoring as well as implicit and explicit user feedback analysis are fundamental tools for runtime adaptation [13]. Regarding the latter, the use of **chatbots** to process user inquiries and feedback is recently growing at exponential levels, especially in specific domains like healthcare or business [2]. They offer not only the potential of collecting and processing natural language feedback at runtime from users to extend missing context information [11], but also to extend the user experience through a smart, human-like conversational interface they are actually familiarized with, which has been proved to be an effective strategy for increasing user engagement [6].

The potential and combination of the aforementioned strategies in the field of Human-Computer Interaction (HCI) is a subject of broad and current interest. Some examples include implicit and explicit feedback collection in the HCI field for personalized mobile UI adaptation [19] and the integration of deep learning-based chatbots for user feedback collection [12]. However, these approaches cover these strategies partially, or they are focused on a limited object of adaptation.

3 The Framework

3.1 Research method

Following the *Goal Question Metric* (GQM) approach, we state the research objective of this paper as follows:

<i>Analyse for the purpose of with respect to from the point of view of in the context of</i>	semi-automatic adaptation mechanisms enhancing user experience features and qualities of mobile applications users mobile software ecosystems.
---	--

To achieve this goal, our research focuses on the design, development, integration and evaluation of an **end-to-end framework for adaptive mobile applications based on the exploitation of explicit and implicit user feedback**. On the basis of the context described in Section 2, we define these adaptation mechanisms to be supported by our framework as follows:

1. Adapt features and qualities of mobile applications (including client apps, back-end and third-party services) at runtime.
2. Install and remove mobile applications to update users applications portfolio.
3. Update mobile applications to deliver customized solutions through user segmentation.

These mechanisms shall be enacted based on users needs and preferences, as well as additional contextual data. To this end, this framework is designed as the integration and combination of a sub-set of stand-alone software components, including: (1) the design and development of a conversational agent in users smartphones to collect and process users natural language feedback; (2) the integration of implicit feedback collection and context monitoring techniques based on users interaction with their smartphones and their applications; and (3) the combination of explicit and implicit feedback to learn and predict user categories and enactment plans for the adaptation of users applications portfolio.

3.2 Design overview

We illustrate our framework proposal (Figure 2) as the software components and data artefacts required to integrate the aforementioned adaptation mechanisms into an end-to-end cycle as a means of achieving the main goal of this research. The framework is depicted in the context of users and the applications portfolio designed and developed by mobile developers as depicted in Figure 1.

User mobile device. The catalogue of applications installed in a user’s device ($\rightarrow AppsPortfolio$) acts as the bridge between that user’s goals and the set of

features and qualities exposed by those applications. To assist the adaptation mechanisms, we propose a cross-platform mobile application ($\rightarrow ChatbotApp$) designed to (1) collect text/speech natural language data as the result of a smart conversational process ($\rightarrow Request-Message$) through the integration of a chat application interface, and (2) collect contextual data based on the user interaction with the applications portfolio and the mobile device ($\rightarrow Context-Data$).

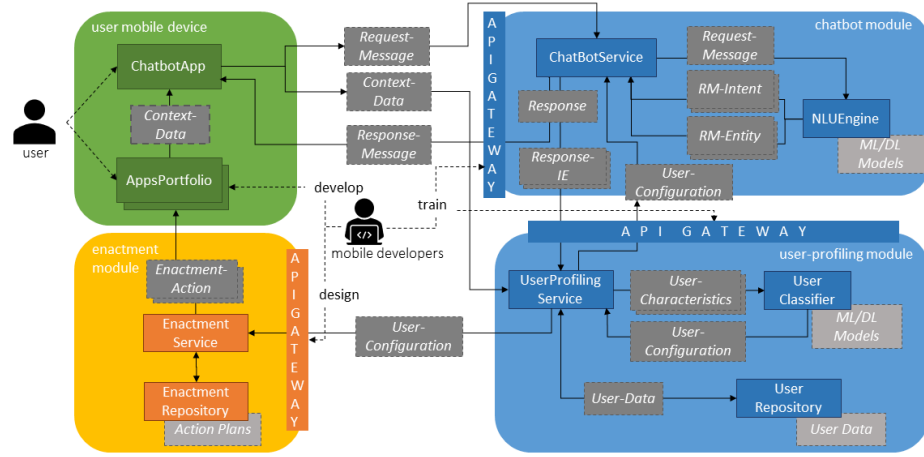


Fig. 2. The architecture of the proposed framework.

Chatbot module. We define the chatbot module as a machine learning software system responsible for processing natural language information ($\rightarrow Request-Message$) to extract interpretable knowledge from explicit user feedback. The technological core of this service ($\rightarrow ChatBotService$) is the Natural Language Understanding (NLU) component ($\rightarrow NLUEngine$), which is responsible for processing natural language data using previously-trained machine/deep learning models. The output of this NLU is two-fold: (1) the user intentions with respect to the applications portfolio to achieve their goals ($\rightarrow RM-Intent$); and (2) the concepts or entities (i.e., applications, features and qualities) these intentions refer to, which require from these adaptations ($\rightarrow RM-Entity$).

These intents and entities feed a response generation algorithm, which generates a response object ($\rightarrow Response$) composed by (1) a natural language response message ($\rightarrow Response-Message$) which is sent back to the user mobile device through the chat application interface, and (2) a set of intent-entity (IE) pairs which relate specific intents with the entities they refer to ($\rightarrow Response-IE$).

User-profiling module. We define the user-profiling module as a machine learning software system responsible for integrating implicit ($\rightarrow Context-Data$) and explicit ($\rightarrow Response-IE$) user feedback to learn and predict user configu-

ration models ($\rightarrow User\text{-}Configuration$), which depict a set of parameterized features and qualities of the user applications portfolio. To compute this configuration, the user-profiling service ($\rightarrow User\text{-}Profiling\text{-}Service$) integrates user feedback with additional data including historic interaction with the mobile applications portfolio and user profile information ($\rightarrow User\text{-}Data$), which is persisted in the system ($\rightarrow User\text{-}Repository$). User feedback and context data are combined to compute a set of structured, deterministic characteristics or metrics ($\rightarrow User\text{-}Characteristics$). These characteristics are the input of a constantly-evolved user classifier ($\rightarrow User\text{-}Classifier$) which uses machine/deep learning models to apply user categorization techniques to predict and learn about user profiles. To address the challenge of evaluating the performance of the chatbot and the user-profiling modules, we suggest the combination of a preliminary generalization stage using open-source, task-oriented dialogue data-sets and a fine-tuning stage using a supervised, domain-specific data-set related to the applications portfolio.

The user-profiling service must include a rule-based approach to determine whether newly computed configurations need from further feedback from the user. This feedback is mandatory to apply critical changes which require from explicit user confirmation (e.g. installing a new application) or from missing knowledge (e.g. uncertainties about a feature). Consequently, user configurations might be sent back to the chatbot service to require this feedback, or they might be sent to the enactment module to apply the adaptation mechanisms.

Enactment module. We define the enactment module as a traditional software system implementing a service ($\rightarrow Enactment\text{-}Service$) responsible for transforming new user configurations of features and qualities to specific adaptation mechanisms or enactment actions ($\rightarrow Enactment\text{-}Action$) to be executed in the user applications portfolio. These actions are eligible through a set of pre-defined adaptation or action plans stored in the system ($\rightarrow Enactment\text{-}Repository$). The service inputs and processes the user configurations predicted by the user-profiling module and retrieves which action plans are required to achieve this new configuration. Each action plan is transformed into this set of enactment actions, which are sent to the user device, closing the end-to-end workflow of the framework.

4 Case-study: The Mozilla Foundation

The Mozilla Foundation³ is a community open source project founded in 1998. Its market-based work, the Mozilla Corporation, is responsible for developing a portfolio of open-source tools for web access and navigation (e.g. the Mozilla Firefox web browser) and a set of open-source mobile applications, including:

- **FirefoxDaylight.** Native Android application of the Firefox web browser.
- **FirefoxFocus.** Private browser for anonymous web navigation.
- **FirefoxNightly.** Developmental channel for users to test new features and provide feedback before publishing them to the stable releases.

³ <https://foundation.mozilla.org/en/who-we-are/>

Firefox implements an Enhanced Tracking Protection (ETP) feature⁴ which allows enabling/disabling web trackers according to the user preferences. This feature allows 4 different configurations: *disabled*, *standard* (blocks all trackers except content tracking), *strict* (blocks all trackers) and *personalized* (customized manually by the user). By default, ETP is set to *standard*.

In this context, we depict three proof-of-concept scenarios to demonstrate how the adaptation mechanisms described in Section 3.1 can be integrated into the framework proposal depicted in Section 3.2. We identify two different users: Alice, a new user in the Mozilla context who has recently added FirefoxDaylight to her device; and Bob, a user with long experience with the Firefox suite.

Scenario A - Adaptation of features and qualities. The user-profiling module detects an uncertainty regarding Alice’s security-related characteristics. The chatbot module sends a message to Alice’s chatbot client app asking her about her security and personal data sharing concerns. Alice makes explicit her reluctance to share any personal data with unknown entities. The chatbot module processes this information and redirects the associated IEs to the user-profiling module, which updates Alice’s persistent data, generating a new user configuration for Alice where the ETP feature in FirefoxDaylight is set to *strict*. Consequently, an enactment action is sent to Alice’s device to apply this change.

Scenario B - Update of users applications portfolio. The user-profiling module updates the classifier models and categorizes Alice under a profile of security-concerned users like Bob. Unlike Alice, Bob has additional applications from the Mozilla portfolio, including the FirefoxFocus anonymous browser. The chatbot module generates a message to inform Alice about this application which might be suitable for users like her. After responding with an explicit confirmation, a new adaptation mechanism is triggered to send a new enactment action to install this application in Alice’s device, and her user profile information is updated according to this change in her applications portfolio.

Scenario C - Customization through user segmentation. Mozilla has published a new functionality F into the FirefoxNightly app to be evaluated by users. After a few weeks of evaluation, the user-profiling module detects that users like Bob have successfully integrated F into their web navigation experience. On the other hand, users like Alice either have reported negative feedback about F or they have not used it at all. Consequently, Mozilla launches a segmented update of FirefoxDaylight including this new functionality, which will only be available for the sub-set of users categorized under Bob’s profile.

These scenarios illustrate our framework as a holistic approach to adapt the Mozilla applications portfolio to their users by simplifying the interaction

⁴ <https://support.mozilla.org/en-US/kb/enhanced-tracking-protection-firefox-android>

required from them. The approach benefits from implicit and explicit feedback to learn about users and adapt their applications towards a better user experience.

5 Future work and conclusions

Further research should be primarily devoted to the following tasks:

- Extend the scope of the depicted adaptation mechanisms that can be integrated into the proposed framework (e.g., software evolution).
- Review challenges from the chatbot domain, with emphasis on (1) analysing sociolinguistic features to increase user engagement and acceptance, and (2) comparing different strategies concerning the integration of a general-purpose NLU Engine with a fine-tuning process for task-specific purposes.
- Design and execute feature extraction techniques to identify relevant user characteristics for an accurate categorization in the context of this research.
- Design a domain-specific language to describe the mapping between user characteristics, user configurations, action plans and enactment actions.
- Research, discuss and elaborate on data collection and data management challenges for the development of the chatbot module and the user-profiling module, as well as for training, testing and evaluating the framework.

Our proposal aims at assisting users to achieve their goals through advanced feedback collection and analysis techniques. Among these, integrating a smart conversational process into the user’s device and using machine/deep learning model for user categorization and segmentation are fundamental to guarantee the novelty of the solution. The scenarios of the Mozilla case-study give a glance at the potential application of our framework to real business cases. Ultimately, these preliminary results lay the groundwork for future research regarding state-of-the-art analysis and initial proof-of-concept development tasks towards a better user experience in mobile software ecosystems.

Acknowledgments

This work has been partially supported by AGAUR, code 2017-SGR-1694. The corresponding author gratefully acknowledges the Universitat Politècnica de Catalunya and Banco Santander for the financial support of his predoctoral grant FPI-UPC.

References

1. Bahia, K., Delaporte, A.: The state of mobile internet connectivity report 2020 - mobile for development (2020), <https://www.gsma.com/r/somic/>
2. Bernardini, A., Sônego, A., Pozzebon, E.: Chatbots: An analysis of the state of art of literature (2018)

3. Braham, A., Buendia, F., Khemaja, M., Gargouri, F.: User interface design patterns and ontology models for adaptive mobile applications. *Personal and Ubiquitous Computing* (2021)
4. Brun, Y., et al.: *Software Engineering for Self-Adaptive Systems*, chap. Engineering Self-Adaptive Systems through Feedback Loops (2009)
5. Chen, Y., Zha, M., Zhang, N., Xu, D., Zhao, Q., Feng, X., Yuan, K., Suyu, F., Tian, Y., Chen, K., Wang, X., Zou, W.: Demystifying hidden privacy settings in mobile apps. In: *2019 IEEE Symposium on Security and Privacy (SP)* (2019)
6. Dev, J., Camp, L.J.: User engagement with chatbots: A discursive psychology approach. In: *Proceedings of the 2nd Conference on Conversational User Interfaces. CUI '20*, New York, NY, USA (2020)
7. Grua, E.M., Malavolta, I., Lago, P.: Self-adaptation in mobile apps: a systematic literature study. In: *2019 IEEE/ACM 14th International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS)* (2019)
8. Jasberg, K., Sizov, S.: Human uncertainty in explicit user feedback and its impact on the comparative evaluations of accurate prediction and personalisation. *Behaviour & Information Technology* (2020)
9. Kemp, S.: *Digital 2020: Global digital overview – global digital insights* (2020), <https://datareportal.com/reports/digital-2020-global-digital-overview>
10. Maia, V., da Rocha, A., Gonçalves, T.: Identification of quality characteristics in mobile applications. In: *CIBSE* (2020)
11. Martens, D., Maalej, W.: Extracting and analyzing context information in user-support conversations on twitter. In: *2019 IEEE 27th International Requirements Engineering Conference (RE)* (2019)
12. Nivethan, Sankar, S.: Sentiment analysis and deep learning based chatbot for user feedback. In: *Data Engineering and Communications Technologies* (2020)
13. Oriol, M., Stade, M., Fotrousi, F., Nadal, S., Varga, J., Seyff, N., Abello, A., Franch, X., Marco, J., Schmidt, O.: Fame: Supporting continuous requirements elicitation by combining user feedback and monitoring. In: *2018 IEEE 26th International Requirements Engineering Conference (RE)* (2018)
14. Orsini, G., Bade, D., Lamersdorf, W.: Cloudaware: A context-adaptive middleware for mobile edge and cloud computing applications. In: *2016 IEEE 1st International Workshops on Foundations and Applications of Self* Systems (FAS*W)* (2016)
15. Picco, G.P., Julien, C., Murphy, A.L., Musolesi, M., Roman, G.C.: Software engineering for mobility: Reflecting on the past, peering into the future. In: *Future of Software Engineering Proceedings*. New York, NY, USA (2014)
16. Qian, W., Peng, X., Wang, H., Mylopoulos, J., Zheng, J., Zhao, W.: Mobigoal: Flexible achievement of personal goals for mobile users. *IEEE Transactions on Services Computing* (2018)
17. Shafiuzzaman, M., Nahar, N., Rahman, M.R.: A proactive approach for context-aware self-adaptive mobile applications to ensure quality of service. In: *2015 18th International Conference on Computer and Information Technology* (2015)
18. Yang, Z., Li, Z., Jin, Z., Chen, Y.: A systematic literature review of requirements modeling and analysis for self-adaptive systems (2014)
19. Yigitbas, E., Hottung, A., Rojas, S.M., Anjorin, A., Sauer, S., Engels, G.: Context- and data-driven satisfaction analysis of user interface adaptations based on instant user feedback. *Proc. ACM Hum.-Comput. Interact.* (2019)