Lacomo: A Layer to Develop Collaborative Mobile Applications

Mauro Carlos Pichiliani [1], Prasun Dewan [2], Celso Massaki Hirata[3]

[1] IBM Research AI
São Paulo, Brasil

[2] Department of Computer Science – University of North Carolina-Chapel Hill
North Carolina, U.S.A.

[3]Department of Computer Science – Instituto de Tecnologia da Aeronáutica
Sao Paulo, Brasil

mpichi@br.ibm.com, dewan@cs.unc.edu, hirata@ita.br

Abstract. *Nowadays, there is little support for developers to transform single user applications to collaborative ones in the mobile domain. We present Lacomo, a new software layer to build collaborative mobile applications with accessibility, screen sharing, and application rewriting technologies that reduce costs to prototype collaboration features, thereby increasing the range of supported applications without deep application knowledge. We compare it to an ad hoc approach. Users using Lacomo performed a testing task faster, with less effort and errors at a higher completion time.*

1. General Information

Mobile applications are a major force behind the explosive growth of mobile devices. While they greatly extend the functionality of those devices, they also open opportunities to enable collaboration, especially with applications familiar to users.

Leveraging single-user commercial systems to multi-user collaboration has been a persistent research topic, since most applications do not have collaborative features. Also, mobile apps such as *Draw Something* achieved high popularity leading us to conjecture that modify and adapt existing mobile application to support synchronous collaboration has potential to reach many users. However, related research academic work and the state of the practice approaches pay little attention to the mobile domain, and the approaches that do so either focus in resources that demand high development effort or do not cope with specific characteristics of mobile devices.

To overcome the above limitations, we propose a Layer to develop collaborative mobile applications (Lacomo). Inspired by operating system's accessibility layers, our approach has the goal to promote collaboration through data sharing in user interface widgets found in existing applications, such as text box and buttons. This is a contrast to traditional approaches to alleviate development through source code modification, component placement, or framework reuse. We envisioned Lacomo as a mobile prototyping approach, which means its usage should be aligned with the states of software development that address collaborative requirements into their projects either a

single-user application that will receive collaboration features or a new application build from the start.

Lacomo is designed to be built on top of an API that capture user events, such as the accessibility API, thus any existing application that has widgets compatible with accessibility interfaces can benefits from its features. The services developed with Lacomo can access events and data associated with elements of the UI interface without application source code in the same way as any accessibility service. Additionally, since Lacomo services can access the Operational System (OS) System log, every communication from the application to the OS and vice-versa is available to the Lacomo layer. The hierarchy of objects that corresponds to UI is available to the developer using callbacks and every pixel shown is reachable due to screen sharing technology. These features achieve a high level of code reuse since they are available at run time and do not require the source code.

We conducted a user study to compare the Lacomo implementation with the *ad hoc* to provides non-WYSIWIS collaboration in an existing mobile application. Formally, this experiment aims to collect quantitative and qualitative evidence to verify the following hypothesis: *Does the use of Lacomo require less effort to modify an existing mobile application than the ad hoc technique?*

20 participants (2 female, 18 male), with age ranging from 22 to 53, participated in the experiments that recorded Time, LOC (Lines of Code), LOC divided by time, Calories, Mouse movements and Save events (Figure 1.
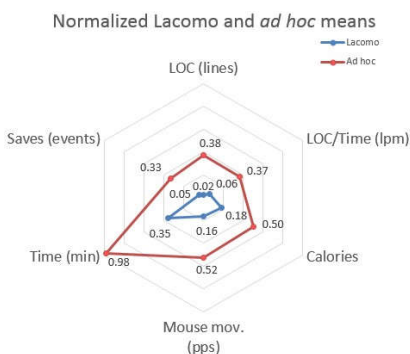

Figure 1. Normalized values of the effort metrics.

The effort required to change a mobile app to support collaborative requirements using the event notification approach compared to *ad hoc* implementation has lower effort metric values. We compared Lacomo with the *ad hoc* approach because to date the is no other automatic approach to convers single-user applications to collaborative ones in the mobile context. Also, the result of the analysis of the changes made by the user support the fact that the complexity is lower than the *ad hoc* approach.

In general, participants commented that the Lacomo approach was interesting. When asked about the difficulty they faced, participants of the *ad hoc* group reported problems to understand the application code and to capture events. These findings were obtained from the analysis of the qualitative data gathered from pos-task interviews.