

Challenges in Android Wear Application Development

Jukka Ahola^(✉)

VTT Technical Research Centre of Finland LTD, Oulu, Finland
jukka.ahola@vtt.fi

Abstract. The research focus of this paper was to highlight current issues in wearable application development in Android Wear platform. The findings of the experimental research were discovered during the software development of a smart watch application, where the requirements and design were limited with the documented best practices. The results identify three issues, which should be addressed in the Android Wear platform roadmap, to better address the wearable application needs. The issues discussed are better wearable internet connectivity, virtual button support for watch faces, and software configurable language support for voice input. We propose those features to be added to the wearable platform roadmap to better support the global wearable business application ecosystems.

Keywords: Android · Google · Smart watch · Voice input · Wearable

1 Introduction

Recently, there has been wide interest in wearable technologies, where the leading global hardware manufacturers have announced their smart watch products in major technology events and conferences. The smart watch platforms are currently evolving in order to meet the usability and aesthetics expectations. [1,2].

However, recent experiments in our application development suggest that the Android Wear software platform [3] does not adequately address some of the usability features needed for smart wearable applications. Android Wear software platform and ecosystem is evolving and is backed by strong industrial partner support, but in its current form it is limiting the user scenarios the application developers are able to pursue.

The purpose of this paper is to highlight selected features needing improvement in the current Android Wear software platform. In this paper, we present three main issues in the wearable development experienced in our work and suggest solutions to overcome them. The rest of the paper is structured as follows: First, we describe what methods were used to identify the usability issues. Then we describe the findings and workarounds to the issues. We discuss the implications of the findings and propose a better way to handle the issues. Then we discuss the related work addressing similar topics and issues. Lastly we draw the conclusions and vision future work to this topic. The experimental results show that the current workarounds to the Android Wear platform address the issues, but require training and instructions for the designed user scenarios. The Android developer issues are discussed in specific developer forums [4,5].

2 Methods

The work was performed in 2014 in Finland. An Android watch face application was developed for the LG G Watch R smart watch paired with a Nexus 5 mobile. The development environment was Android Studio version 1.0 with Android SDK API version 24. LG G Watch R has a round display, which requires special attention when designing the graphical layouts.

3 Results

The Android Wear software platform does not support direct internet access from the watch [3]. The watch is connected to a smart phone through Bluetooth. The companion mobile application or service has to function as a proxy between the wearable application and the internet services. If the paired mobile application is not available for some reason, the user may not be aware of it and the wearable application is not getting the data from the web service. Wearable applications also rely on many services provided by the GoogleApiClient [3].

Android Wear compatible watches only have one physical button, and it is used to turn off the display or to return to the default watch face mode. Wear application can have buttons and other user interaction components in the application interface. However, the Android Wear application deactivates to the default power save mode after a short idle time. Watch face application is a long running application by default and is designed to display the time as shown in Figure 1. Android Wear takes over the touch events for watch face applications, preventing any action buttons to be triggered from the touch screen.

Android Wear supports receiving voice input from notifications. This requires the user to swipe the proper notification that triggers a separate activity to handle the voice input functionality. The speech recognition utilizes the the microphone on the wearable device, and translates it to text by using the Google cloud services. The voice input uses the language options configured and supported by the mobile phone. For instance, if the mobile phone default primary language is set to Italian, the wearable will attempt to recognize the speech in Italian. However, if you intend to override the language option programmatically from the Android Wear-application, the mobile phone will ignore it and use the default language options instead.

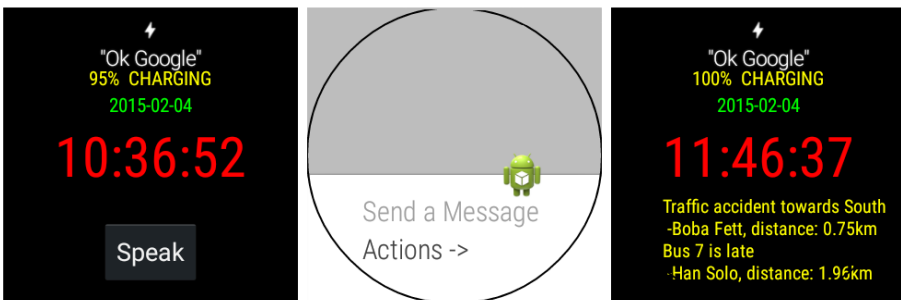


Fig. 1. Watch face application with a button and a voice input notification functionality

4 Discussion

The goal was to implement a wearable application, which is easily and quickly available, and able to send short spoken messages as text to the server. The application must be available with one touch. Watch face application was selected, because it activates itself with the push of a button and stays on until the user selects a different watch face application. Standard Android Wear applications can be configured to be long running applications by utilizing the wake lock feature. However, the application requires multiple swiping motions with accuracy to be activated, which can be troublesome in situations requiring attention elsewhere. The spoken language could not be changed in the watch face application due to the limitations in touch events and software configured language settings. The wearable application relies on the languages options configured on the mobile device. Currently the latest Android speech recognition supports two language options at the same time, where one language is selected as the primary language. This is a useful feature for localization, but cannot combine the two languages simultaneously, when you need to spell foreign street addresses in your speech. The wearable application currently has no understanding of the internet. It relies on the Bluetooth connection with the paired mobile companion application to delegate any web service requests and responses through mobile internet connectivity. The mobile application receives URL-addresses from the wearable application, and creates web requests with asynchronous responses back to the wearable application.

The following features would greatly benefit the future Android Wear platform:

- Wearable API that allows URL connections either directly or transparently through the connected mobile phone.
- Ability implement virtual software buttons to watch face applications for user interaction.
- Ability to programmatically control the language preference for voice input from the watch.

With the above features, the applications could be easily accessible in situations where the information needs to be accessed with limited eye-contact and touch interaction. The wearable application dependency from the companion mobile application would decrease with the support of internet networking library, making the development more straight forward and easier to maintain. Internationalization and localization would also be better supported, when the control is given to the wearable device on the wrist instead of the mobile device in the pocket. Watch Face applications for Android Wear could be more interactive with the support of virtual buttons, allowing users to trigger functionality directly from the watch face screen.

5 Conclusion and Future Work

Android Wear is a fascinating software platform capable of exchanging information in new ways. This paper identifies and discusses three important features currently

missing in Android Wear platform that would greatly benefit wearable software developers. Direct internet connectivity, virtual button support for watch faces, and voice input localization should be considered to be included into the Android Wear platform to make meaningful wearable business applications. We will investigate alternative solutions and best practices to overcome the current limitations on the platform.

Acknowledgements. The described work discussed in this paper, is part of the Connecting Digital Cities activity of 2014, funded by EIT ICT Labs Knowledge and Innovation Community. The author wants to thank the colleagues participating the design and implementation of the system presented in this paper.

References

1. Morris, I.: Forbes (2014). <http://www.forbes.com/sites/ianmorris/2014/12/30/google-needs-to-fix-android-wear-in-2015-to-fend-off-the-apple-watch/> (accessed February 4, 2015)
2. Fedewa, J.: Phandroid (2014). <http://phandroid.com/2014/11/06/7-things-i-hate-about-android-wear/> (accessed February 4, 2015)
3. Android.com. <https://developer.android.com/training/building-wearables.html> (accessed February 4, 2015)
4. Stack Overflow. <http://stackoverflow.com/questions/tagged/android-wear> (accessed February 4, 2015)
5. Android Wear Developers. <https://plus.google.com/communities/113381227473021565406> (accessed February 4, 2015)