*Health Informatics Meets eHealth*
*D. Hayn and G. Schreier (Eds.)*

*doi:10.3233/978-1-61499-759-7-336*

# Bluetooth Low Energy Peripheral Android Health App for Educational and Interoperability Testing Purposes

Matthias FROHNER[a,1], Philipp URBAUER[b] and Stefan SAUERMANN[a]
[a]*University of Applied Sciences Technikum Wien, Department of Biomedical, Health and Sports Engineering, Wien, Austria*
[b]*University of Applied Sciences Technikum Wien, Department of Information Engineering and Security, Wien, Austria*

**Abstract.** Based on recent telemonitoring activities in Austria for enabling integrated health care, the communication interfaces between personal health devices (e.g. blood pressure monitor) and personal health gateway devices (e.g. smartphone, routing received information to wide area networks) play an important role. In order to ease testing of the Bluetooth Low Energy interface functionality of the personal health gateway devices, a personal health device simulator was developed. Based on specifications from the Bluetooth SIG a XML software test configuration file structure is defined that declares the specific features of the personal health devices simulated. Using this configuration file, different scenarios are defined, e.g. send a single measurement result from a blood pressure reading or sending multiple (historic) weight scale readings. The simulator is intended to be used for educational purposes in lectures, where the number of physical personal health devices can be reduced and learning can be improved. It could be shown that this simulator assists the development process of mHealth applications by reducing the time needed for development and testing.

**Keywords.** Mobile Applications, Software Validation, Telemonitoring

## 1. Introduction

Digital communication interfaces are a core requirement for electronic exchange and sharing of data between software applications. In the context of health telematics this can be observed within the introduction and implementation of electronic health records systems (EHRs), which enable health professionals to share clinical information independent of time and place. These electronic health records systems are supposed to be used mainly within a professional domain and with requirements and challenges characteristic for this domain. Nevertheless, due to the needed integration of software solutions provided by multiple software vendors, the specification of the communication interfaces is crucial. For the electronic health records system in Austria those specifications are based on profiles defined by IHE [1] and are adapted and provided by the ELGA GmbH [2]. Beside the Austrian electronic health record system being developed, efforts are made by the Austrian Federal Ministry of Health and Women's Affairs to design an architecture for telemonitoring applications [3]. For telemonitoring

[1] Corresponding Author: Matthias Frohner, Department of Biomedical, Health and Sports Engineering, University of Applied Sciences Technikum Wien, Höchstädtplatz 6, 1200 Wien, Austria, E-Mail: matthias.frohner@technikum-wien.at.

applications the personal health devices, that a patient/user is supposed to operate at home, will represent the source of the data that should be exchanged. Therefore, those devices need to implement a communication interface that send measured data to a personal health gateway (e.g. the patient's/user's smartphone) in order to forward this data to centralized telemonitoring services. For this interface between personal health devices and smartphones the Personal Connected Health Alliance [4] (cofounded by the Continua Health Alliance) defines the interfaces' implementation according to a set of communications standards. For the time being, the Continua Design Guidelines for Personal Health Device Interfaces [5] provides specifications for

- ISO/IEEE 11073-20601 (X73 PHD) over
    - NFC,
    - Bluetooth Basic Rate / Enhanced Data Rate,
    - USB, and
    - ZigBee, or
- Bluetooth Low Energy

as the communication protocol and transport channels respectively.

Having the specifications in place, personal health devices manufactures and software applications implementers can implement those interfaces and further on register for certification by the Personal Connected Health Alliance.

For the development of a personal health gateway application facilitating an Bluetooth Low Energy interface for data communication with personal health devices an easy to use test engine is needed in order to re-run test cases where medical data should be received from a personal health device. In cases where a body weight measurement should be transmitted this can be achieved by a weight scale device quite convenient. However, when a blood pressure reading or a blood glucose reading shall be transmitted, recurring measurement cycles are time consuming and are not recommended from a medical point of view. A similar challenge is considered for educational purposes, where students should be taught the basics of medical device communication and are supposed to implement Android applications that can receive data from personal health devices. Providing a sufficient hardware stock of personal health devices for lectures and software development project is not always possible. In both cases, testing and training, a software-based personal health device simulator can reduce development time and improve quality by enabling implementers or software test engineers to use a simulated blood pressure monitor or weighting scale to transmit data. The design and the development of such a personal health device simulator is an objective of the funded project INNOVATE [6]. Within the project an innovation laboratory implementing eHealth technologies will be established, providing knowledge and resources for implementation and testing of secure and interoperable software solutions. For interoperability testing, i.e. testing if data can be communicated to, and interpreted by the receiving software component, the use of simulators seems appropriate. The IHE provides within its test bed "Gazelle" multiple simulators to be used for interoperability testing [7].

This paper defines and reports the requirements for test case configuration and the implementation of an Android application that simulates personal health devices (see Figure 1). The test case configuration designed, holds the relevant data for various test cases. This configuration structure considers the different information models that are the bases for different personal health device classes; a single valued weight reading vs. a related set of values for systolic, diastolic, and mean blood pressure. The configuration

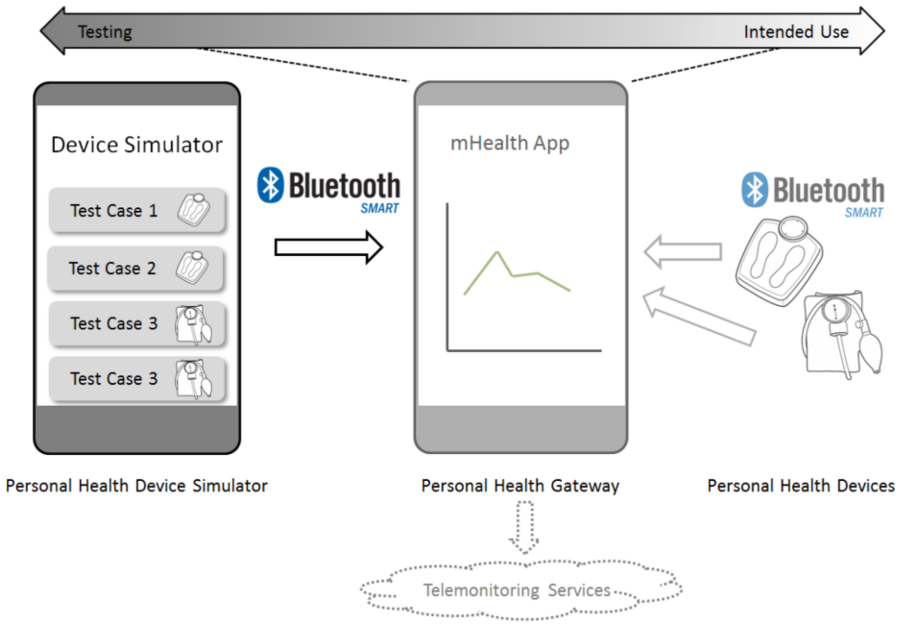


**Figure 1:** Personal health device simulator application simulating personal health care devices for interoperability testing and educational purposes

additionally considers the option that personal health devices are sending one measurement data set after the measurement cycle or sending multiple (historic) measurements sets when data have been stored in the device's internal storage. In addition, further device capabilities, like support of a real-time clock or the capability to manage multiple users should be stated in the configuration file.

This configuration provides the possibility to adjust and define software test cases without the need to adapt the code of the application. Having this test case configuration feature in place, it allows to extend the software test portfolio over time, i.e. configurations for new personal health devices can be added in the configuration file resulting in new test cases that can be executed using the graphical user interface of the application.

One requirement for the smartphone, where the personal health device simulator application is executed, is that this phone supports the capability to act as a Bluetooth Low Energy "data source". The Bluetooth SIG uses the term *Bluetooth Low Energy Peripheral* for such devices. Android introduced this feature with Android 5.0 [8] however there is strong evidence that the phone's firmware must support Bluetooth Low Energy Peripheral Mode as well and that the latter is not available on a broader number of Android phones available at the moment [9].

## 2. Methods

First, a suitable Android smartphone with enabled Bluetooth Low Energy Peripheral mode is selected. In order to validate if a smartphone supports this feature specific applications from Google's Play Store can be used (e.g. [10, 11]). Android smartphones and tablets at hand, running on Android 5.0 or higher have been tested for this support:

- HTC Google Nexus 9, 16 GB WiFi, Android 7.1.1 (Build-Number NMF26F)
- OnePlus 3, A3003, Android 7.0 (Build-Number ONEPLUS A3003_16_170114)
- Honor 8, FRD-L09, Android 6.0 (Build-Number FRD-L09C432B131)

From the three tested devices only the Honor 8, running on Android 6.0, does not support Bluetooth Low Energy Peripheral mode.

For the structure of the test configuration file a XML structure is defined, that can hold all the relevant information for different test cases. Test cases shall enable a user to send multiple simulated measurements with different timestamps for simulated blood pressure monitoring devices, weighting scales, and blood glucose meters. In order to identify the needed data for the configuration file the corresponding Bluetooth Low Energy service specifications from the Bluetooth SIG are studied [12]. In addition, requirements for Bluetooth Low Energy profiles set by the Continua Design Guidelines are considered.

A first prototype implementation is implemented in Android Studio (API 24, Android 7.0) and the capabilities of the developed application simulating personal health devices is tested against an Android application based on former developments [13].

## 3. Results

Based on the investigation on the supported content defined in the Bluetooth Generic Attribute Profile (GATT) services

- Device Information,
- Battery Service,
- Weight Scale,
- Blood Pressure, as well as
- Glucose

and the corresponding characteristics, a XML configuration structure has been designed. Basic device information - stated in the Device Information service - overarching different device classes are reflected by a set of common XML attributes. Informational content that is only available for a specific device class is mapped to a "key-value"-like representation in the XML (see Figure 2).

Moreover, the availability of certain informational objects, like support of BMI for weight scales, is implemented by the Bluetooth SIG by setting a flag in the characteristics features list. The XML structure does not use this explicit flagging but provide this implicitly when a corresponding key-value pair exists. The Android simulator application will then sets the flag in the corresponding feature list.

After the Bluetooth pairing has been finished, the simulator application is loading the configuration file and based on the number of defined test cases (*TestCase* XML tag) the user interface will be populated with buttons for test case execution. The read-characteristics (e.g. device information) that can be derived from the configuration file will then be accessible for external reading and subsequently indication-characteristics (e.g. blood pressure measurements) will be available.

```xml
<TestCases>
    <TestCase tId="1" profile="WS">
        <DeviceInformation manufacturerString="ManufacturerXY" modelNumberString="ModelAB"
         serialNumberString="123456789" hardwareRevisionString="asdf"
         softwareRevisionString="12.34" systemId="dev123"
         regulatoryCertificationData="65551"/>
        <Measurements>
            <Measurement mId="1">
                <flag type="timeStamp" value="20160115121500"/>
                <flag type="userId" value="1"/>
                <flag type="BMI" value="24.1" unit="1"/>
                <flag type="mass" value="83.4" unit="kg"/>
                <flag type="height" value="1.86" unit="m"/>
            </Measurement>
        </Measurements>
    </TestCase>
    <TestCase tId="2" profile="BP">
        <!--
        ....
        -->
    </TestCase>
</TestCases>
```


**Figure 2:** XML-example for a weighting scale test case

## 4. Discussion

The XML-configuration has been used to simulate

- a blood pressure monitor device sending one set of measurements (systolic-, diastolic-, and mean blood pressure as well as pulse),
- a weight scale device sending three different body masses at three different points in time (transfer of historic data), and
- a blood glucose meter sending five different glucose measurements recorded over time.

This configuration has been provided to the developed Android application. The application has been executed on HTC's Nexus 9 tablet and the capability of simulating devices has been validated against health application from Google Play Store (e.g. [14]) and applications developed in the context of the research project.

The implemented personal health device simulator app seems to be an appropriate tool that can be used for testing during the development phases, because the test case configuration can easily be adapted to the very needs of software developers. The simulator app can be used for the validation of BLE interface capabilities of health- and fitness software applications under development that are supposed to communicate with personal health care devices. Since the simulator is designed for interoperability tests and acts merely as an independent test stub there are no further constrains for the overall software test methodology of a software manufacturing company. If a Continua Certification for the personal health gateway is intended, the Continua Test Tool documentation [15] declares the use of the "Profile Tuning Suite/Bluetooth Developer Studio Radio Module" [16] for simulating a personal health device.

One issue that is not addressed with the used XML configuration structure is the circumstance that it cannot be coded that certain *flags*-attributes are mandatory, i.e. data shall be available according service and characteristic specifications from Bluetooth SIG. This logic is currently implemented in the Android application. Another future possibility would be not to use the generic key-values approach but to introduce more specific XML elements for the different device classes and providing a XML schema file. Another approach would be the transfer of the logic from the application into a XML schematron file containing the content and validating the used configuration file. In general, a trade-off between increasing the complexity of the configuration file and amount of logic that is hard-coded within the applications has to be considered. When the code is kept very generic, more knowledge is needed for setting up a configuration file that conforms to the requirements stated by Bluetooth SIG, i.e. required key-values for GATT services and dependencies between different attributes need to be stated in the configuration file itself. Nevertheless, the test engineer should be enabled to extend the simulator app with new test cases by adding additional test case specifications in the configuration file. However, extending the simulator app with new personal health devices to be simulated is not possible in the current set-up. This is due to the fact, that needed logic for devices is currently coded in the application itself. In future this logic might be transferred to the configuration file, but this will require sufficient knowledge of the test engineers maintaining the configuration.

The application is intended to be used in lectures in the upcoming study semesters where students are supposed to implement software projects showing the communication flow from the personal health device towards an electronic health record system. Based on the experiences, further improvements and features might be implemented. Feedback and experiences gathered during the lectures is expected to influence the decision whether to increase the complexity of the configuration file allowing the definition of new health devices to be simulated, or to stick to a simple configuration file where additional test cases can be defined rather easily. The latest version of the personal health device simulator application will be available for public download on the project homepage.

Software testing tools like the presented personal health device simulator application are essential artefacts to reduce development time and to improve the quality of software products, in terms of interoperability, and might play a role in software certification processes in future.

## Acknowledgment

This project (project-number 19-06) is funded by the City of Vienna Municipal department 23 “Economic Affairs, Labour and Statistics” within the program “Research at Viennese Universities of Applied Sciences.

## References

[1] IHE International Inc., “Integrating the Healthcare Enterprise.” [Online]. Available: http://www.ihe.net/. [Accessed: 30-Jan-2017].

[2] ELGA GmbH, “ELGA - die elektronische Gesundheitsakte.” [Online]. Available: http://www.elga.gv.at/. [Accessed: 30-Jan-2017].

[3] S. Sauermann and I. Weik, “eHealth Applications in Austria: Telemonitoring,” in 7. Nationaler Fachkongress Telemedizin.

[4] Personal Connected Health Alliance, “Personal Connected Health Alliance.” [Online]. Available: http://www.pchalliance.org/. [Accessed: 30-Jan-2017].

[5] Personal Connected Health Alliance, “H . 811 Personal Health Devices Interface Design Guidelines, Version 2016 (August 4, 2016),” 2016.

[6] UAS Technikum Wien, “Innovate.” .

[7] IHE International Inc., “Simulators.” [Online]. Available: https://gazelle.ihe.net/content/simulators. [Accessed: 10-Mar-2017].
[8] Android Developers, “Android 5.0 APIs (API Level 21).” [Online]. Available: https://developer.android.com/about/versions/android-5.0.html. [Accessed: 30-Jan-2017].
[9] Radius Networks, “Android Beacon Library.” [Online]. Available: http://altbeacon.github.io/android-beacon-library/. [Accessed: 03-Feb-2017].
[10] Hochi, “BLE peripheral check.” Google Play Store.
[11] LLC U7, “BLE Peripheral Detection.” Google Play Store.
[12] Bluetooth SIG Inc., “GATT Services.” [Online]. Available: https://www.bluetooth.com/specifications/gatt/services. [Accessed: 02-Mar-2017].
[13] M. Frohner et al., “Development of an android app in compliance with the continua health alliance design guidelines for medical device connectivity in mhealth,” Biomed. Tech., vol. 57, no. SUPPL. 1 TRACK-N, pp. 997–999, Sep. 2012.
[14] MedM Inc, “MedM Weight.” 2012.
[15] AT4 wireless S.A.U. and Personal Connected Health Alliance, Continua Test Tool DG2016 PHD Interface Test Tool Usage Document, Version 1.9, 2016-09-20. 2016.
[16] BluetoothStore, “The Profile Tuning Suite/Bluetooth Developer Studio Radio Module (BR/EDR and LE).” [Online]. Available: https://bluetoothstore.org/. [Accessed: 02-Mar-2017].