eHealth2014 – Health Informatics Meets eHealth
A. Hörbst et al. (Eds.)

doi:10.3233/978-1-61499-397-1-9

Framework for Near-Field-Communication-Based Geo-Localization and Personalization for Android-Based Smartphones—Application in Hospital Environments

Philipp MENG[a,b], Karsten FEHRE[c], Andrea RAPPELSBERGER[d], and Klaus-Peter ADLASSNIG[c,d,1]
[a] *AKh Linz GmbH, Linz, Austria*
[b] *Philipp Meng, Individual Enterprise, Linz, Austria*
[c] *Medexter Healthcare GmbH, Vienna, Austria*
[d] *Section for Medical Expert and Knowledge-Based Systems, Center for Medical Statistics, Informatics, and Intelligent Systems, Medical University of Vienna, Vienna, Austria*

Abstract. Various applications using near field communication (NFC) have been developed for the medical sector. As a method of short-range wireless contact-driven data transfer, NFC is a useful tool in medicine. It can be used to transfer data such as blood pressure, control adherence to medication, or transmit *in vivo* data. The first proposed general framework uses NFC as a mechanism for indoor geo-localization in hospitals. NFC geo-localization is economical compared to classical concepts using indoor GPS or WLAN triangulation, and the granularity of location retrieval can be defined at a tag level. Using this framework, we facilitate the development of medical applications that require exact indoor geo-localization. Multi-user Android systems are addressed in the second framework. Using private NFC tags, users are able to carry on their personal settings for enabled applications. This eliminates the need for multiple user accounts on common Android devices, improves usability, and eases technical administration. Based on the prototypes presented here, we show a novel concept of using NFC-enabled Android devices in hospital environments.

Keywords. Medical informatics applications, mobile health, telemedicine, cellular phone, near field communication, geographical locations.

1. Introduction

Mobile applications have significant benefits in everyday life. The number of available applications in medical practice is very diverse and is still growing. Applications exist for telemonitoring-supported treatment of chronic heart failure [1], therapy management of diabetes mellitus [1, 2], and documentation of medication [3]. The costs and the organizational effort of adoption may hinder widespread deployment of

[1] Corresponding Author: klaus-peter.adlassnig@meduniwien.ac.at

such applications in clinical practice. Multi-user systems with personalized settings, allowing several users to utilize the same hardware, may resolve the above-mentioned obstacles. In addition, some of the available applications could be substantially improved for medical practice, if it were possible to determine their current location (the actual ward or room in a hospital). With this information it would be possible to display only relevant information for the room the doctor has just entered during his/her visit.

We propose multiple components based on near field communication (NFC) technology for mobile devices, which could be used in any mobile application for indoor geo-localization and application personalization.

Any wireless technology may be used for indoor geo-localization. Although rather costly, indoor global positioning systems (GPSs) are currently the gold standard for this application. Radio-frequency-identification (RFID)-based location systems are popular in logistics, and have been rated positively for hospital settings as well [4]. The use of wireless local area networks (WLANs) is very popular because it is based on the pre-existing infrastructure. Bluetooth may also be used to determine the user's position. However, some issues have to be kept in mind. Failures in reception due to the breakdown of a wireless station may lead to false results in localization calculation. Constructional changes influence the reflection of radio waves and may lead to false results as well. For non-critical settings, where 100% accurate positioning is not necessary, these issues might not matter. In a medical setting as in hospitals, these potential errors could lead to critical events.

Personalization of applications is part of the usability of software. Personal customization in the context of multi-user systems is challenging. Various solutions are available at the desktop level. The majority of these employ a user name and a password for authentication, or the use of a personal smart card. All of these solutions offer possibilities of central administration and support. Mobile technologies such as tablets do not permit customization in multi-user environments or easy administration by information technology staff. Android offers out-of-the-box multi-user support with the possibility to grant rights on a granular level [5], but such support for restricted profiles is limited to eight users. Additionally, it is not possible to share a specific set of basic preferences among the applications. These are significant requirements to enable mobile devices with true multi-user support.

NFC is a technology based on a set of standards that allow short-range wireless communication between mobile devices. Since its introduction in 2002 [6], various applications have been developed in the medical sector. The transmission of small quantities of data using a contact-driven approach offers high levels of usability. This facilitates the transfer of medical data such as blood pressure, oxygen saturation, or weight to smart phones for further applications. Even telemedical applications have been tested, such as the transmission of electrocardiographic data and blood pressure data of patients suffering from congestive heart failure [7]. Schreier et al. evaluated the use of NFC in telemedical settings compared to the classical approach using a web interface, and came to the conclusion that NFC enhances compliance [8]. Lantada et al. found NFC to be an attractive aid in the transmission of real-time *in vivo* data [9]. Adherence to medication has also been tested with NFC [9].

2. Methods

We developed two separate applications. The first one—with "nfcgeo" as the working title—implements the geo-localization functionality. The second one—referred to as nfcsettings in this publication—permits the storage of personal settings on NFC tags to enable multi-user support.

Basic requirements for both applications are an NFC-capable Android device and an NFC tag. As a method of short-range data transfer, NFC requires close proximity of the reading device to the NFC tag. The maximum distance between the device and the tag is theoretically limited to ten centimeters; it typically ranges between three and five centimeters. After initiation of data transfer, the Android NFC dispatch system parses the stored information and forwards the information—typically NFC data exchange format (NDEF) records encapsulated in an NDEF message—to applications capable of further processing, based on the multipurpose internet mail extensions (MIME) type of NDEF records encapsulated in the NDEF message.

2.1. NFC Tags

Four different NFC tag types have been defined by the NFC Forum [11]. For our applications, we use type 2 tags. These tags are read-write capable, with an optional read-only setting, and a memory capacity of up to 2 kilobytes. For our prototype, we use NTAG203 tags. These are NFC Forum type 2 compliant tags with 144 bytes of available memory. The memory area is divided into 36 pages with 4 bytes each, and basic security is offered by a unique 7-bit serial number for each tag. The specifications for these tags can be found at [12].

2.2. NDEF Message

We decided to use the NDEF message system because Android provides sophisticated support for NDEF message processing, and implements a dispatch system. An NDEF message is a container for one or more NDEF records; each NDEF record may contain

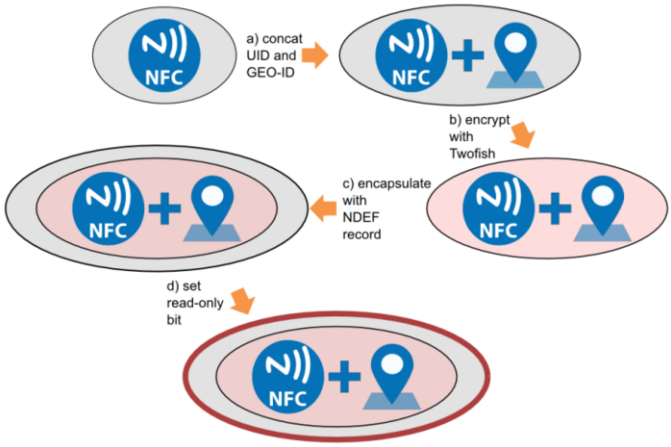

Figure 1. The security setup of nfcgeo.

type, the dispatch system decides on the capable applications for further processing of data in a predefined MIME type or custom application payload. Based on this MIME data. The disadvantage of using NDEF messages in this context is its vulnerability. Theoretically, every application can read the data from the NDEF messages. Therefore, nfcgeo data are stored in encrypted form on NFC tags.

2.3. Nfcgeo Tag Initialization

The tags used in the context of nfcgeo must be marked with correct information about their geographical position. As the tags need to be tamper-proof, the read-only bit has to be set. To ensure that only tags certified for the correct geographical position are in use, we employ a security setup.

When setting up a new tag, the unique identifier (UID) is first read from the tag and then combined with the geographical identifier (Figure 1a) to be used with the specific tag. This string is encrypted using the symmetrical encryption algorithm Twofish (Figure 1b). The encryption key for Twofish is hard-coded inside the program code and not accessible from the outside. The resulting data is encapsulated in a custom NDEF message and written to the tag (Figure 1c). Afterwards the read-only bit is set (Figure 1d). Further modification of the data on the tag is now impossible. All tags prepared as described above may then be positioned at the defined geographical position. To prevent subsequent change of location of tags, security seals should be used to visually demonstrate their trustworthiness.

The geographical position should not be specified in the classical data format with latitude and longitude. String values or a combination of both formats are also permitted.

Figure 2. The GUI of the Android applications. Left and center: nfcgeo, right: nfcsettings.

2.4. The Nfcgeo Application

The functionality of the nfcgeo application for everyday use requires no graphical user interface (GUI). Still, there is a simple GUI (Figure 2, left and center) for the configuration of allowed applications; it may be used by the administrator during the initial setup. To prevent unauthorized changes by users, this GUI is password protected. Additionally, as these settings are stored in Android's preferences system, all these data are encrypted by the Twofish algorithm using a secret encryption key. The GUI can be employed by the administrator to select the applications permitted to access the geographical position. These applications need to have an application programming interface (API) key to access the geographical data. The API key is the private key of an asymmetric encryption algorithm. As the geographical position is encrypted with the public key, other applications cannot use the position without appropriate authorization. This is necessary becauses Chin et al. found that inter-application communication in Android is subject to various types of insecurity [13]. Lastly, the administrator may select one of these applications to be started automatically when an nfcgeo NFC tag is within reach. The entire process is shown in Figure 3.

2.5. The Nfcsettings Application

This application is in a minimalistic GUI, with only one function that permits any new writable NFC type 2 tags to be formatted for use with nfcsettings. The tags can be initialized by the user. Additionally, an introduction to the use of the application is provided in the user interface (Figure 2, right).

Owing to limited storage capacities, a minimalistic concept is required for data storage on NFC tags. In our prototype, we use low-cost tags with an accessible capacity of 144 bytes. We decided to store the data as an array of the char data type. The application context is encoded as a string using 4 bytes of storage; it is decoded with a local lookup table.

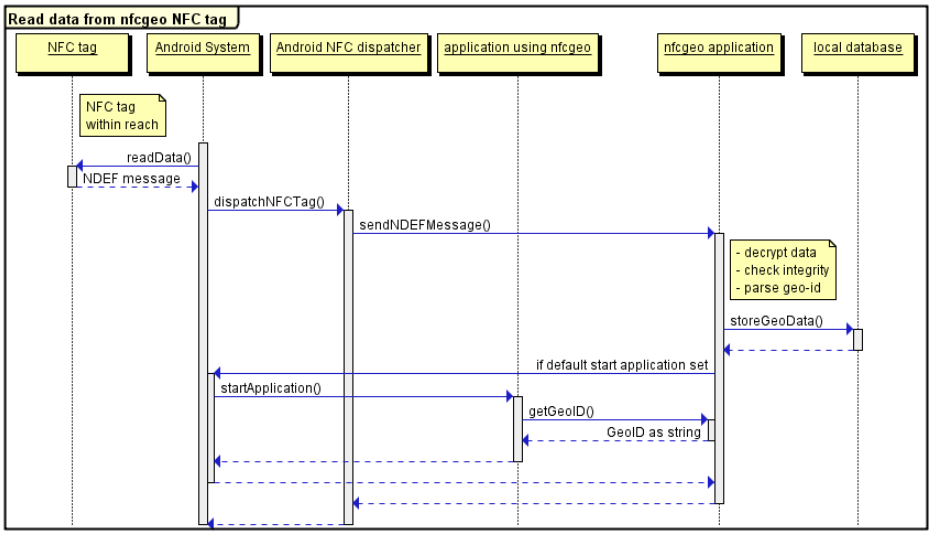

Figure 3. Process started when an nfcgeo NFC tag is within reach.

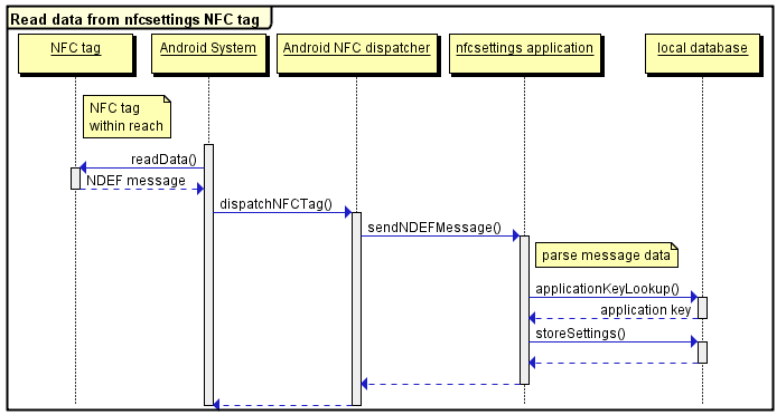

Figure 4. The process of reading data from an nfcsettings NFC tag.

If the tag initialized for nfcsettings is within reach, the data is read. In our case, by defining a custom MIME type, we can be sure that nfcsettings is the application of choice. The message data is parsed, a lookup of the full application qualifiers is performed, and the settings are stored in the local database for further use (Figure 4).

When an application supporting nfcsettings is being started, an intent calling nfcsettings for any available personal preferences is launched. After an internal check, if the application is permitted to fetch settings from nfcsettings, an array of chars specific to this application is returned (Figure 5).

Applications capable of storing preferences in nfcsettings pass this configuration data as an array of the char data type to the nfcsettings application using intents. As nfcsettings do not manipulate these data and the NFC tags have limited storage space, the nfcsettings API specifies a certain size for the data to be stored. This data array is temporarily stored in a local database. When an initialized NFC tag is within reach, the data is written directly to it, without the need of interaction from the user.

When the screen of the device is turned off, the data previously fetched from an NFC tag need to be wiped. We use a threshold of ten seconds for the wipe. The reason for this is usability, as the screen may turn off automatically without the user's intention. In these cases, the user will turn the screen on instantly again. Otherwise, personal NFC tags can be affixed to private items. Besides, the use of NFC-supporting gadgets like NFC key ring pendants or NFC bracelets is also permitted.

3. Results

In the present report, we demonstrate novel applications with NFC, especially in the medical sector. With our two applications, users and application developers may derive exact geo-localization information that may subsequently be processed by other applications, as well as use NFC tags to store personal settings of common Android devices.

With our geo-localization approach, software companies can develop medical applications that depend on the exact position of the user in medical services. Additionally, our nfcsettings application is a prototype for a novel framework of

storing personal settings of applications in the context of multi-user systems. This allows the user to personalize systems in a limited setting, currently because of finite storage on NFC tags.

The desire for these two applications arose during the development of another project for exact geo-localization in hospitals, involving applications with multi-user capabilities.

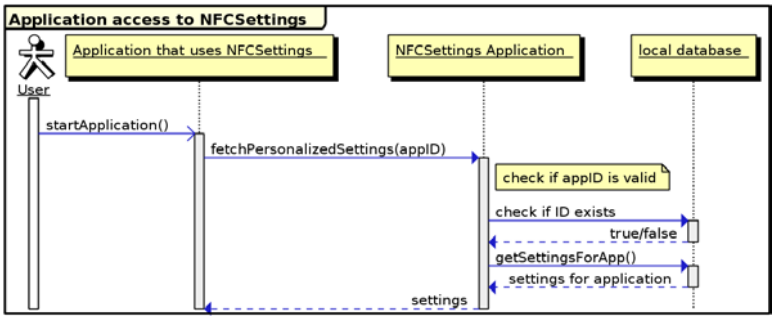

Figure 5. Process of communication between applications and nfcsettings.

4. Discussion

In the past years, NFC has become increasingly popular in the medical sector. Although NFC is used in various applications for easier transfer of patient data, little has been done to evaluate potential enhancements for medical staff through the use of this new technology. Our two applications show two new interesting areas for the use of NFC in the medical context.

With the new framework prototyped in nfcgeo, we offer exact indoor geo-localization to medical smart phone applications. Compared to the use of RFIDs, in NFC the active technology is located in the smart phone. Thus, no communication with a location server is needed to determine the geographical position. Data security is a major concern in the medical setting. We use a layered security concept to increase the validity of the data. The electronic layer involves the encryption of data using a strong symmetric encryption algorithm, including the UID of the NFC tag. The physical layer is the prevention of movement of NFC tags by the use of custom hologram security seals to cover these. The user is able to identify a moved tag because the seal will be broken. Custom hologram stickers make it difficult to counterfeit the seals. The NFC tags are preferably located on immobile items, such as walls or doors. Additionally, as the read-only bit is set, the tags cannot be overwritten. Thus, the data cannot be replaced on previously positioned NFC tags.

The storage of an encryption key hard-coded in the byte code is problematic because the key can be found by de-compilation of the code. However, this has been rendered difficult by calculation of the encryption key at runtime from various constants defined in the program code rather than storing the key as a clear text string in the code.

Nfcsettings offers the use of multi-user environments on mobile Android devices. Using personal NFC tags is a cheap method of storing personal settings and information, and a significant improvement on existing options. In fact, a major

challenge in the future will be the evaluation of possibilities to turn Android devices—mainly tablets—into real multi-user machines, probably with the use of NFC smart cards. As these contain secure elements and have encryption algorithms on board, true authentication and even the storage of credentials will be possible. Additionally, the use of hybrid smart cards—wherein the same secure element is accessible as a smart card to a computer and to mobile devices through NFC—might further enhance usability.

References

[1] J. Morak, G. Schreier, *Mhealth Based on NFC Technology – Preliminary Results from Medium Scale Proof of Concept Projects*, in: G. Schreier, D. Hayn, A. Hörbst, E. Ammenwerth (eds.) Proceedings of the eHealth2012. Österreichische Computer Gesellschaft, Wien, 2012. pp. 131–137.

[2] A. Kollmann, M. Riedl, P. Kastner, G. Schreier, B. Ludvik, Feasibility of a mobile phone-based data service for functional insulin treatment of type 1 diabetes mellitus patients, *Journal of Medical Internet Research* **9**(5) (2007), e36.

[3] M. Schwarz, R. Modre-Osprian, D. Scherr, F. Fruhwald, G. Schreier, *ADOKA: Prototyp eines NFC-basierten mobilen Arzneimittel-Dokumentations-Assistenten*, in: E. Ammenwerth, A. Hörbst, D. Hayn, G. Schreier (eds.) Proceedings of the eHealth2013. Österreichische Computer Gesellschaft, Wien, 2013. pp. 155–160.

[4] A.A.N. Shirehjini, A. Yassine, S. Shirmohammadi, Equipment location in hospitals using RFID-based positioning system, *IEEE Transactions on Information Technology in Biomedicine* **16**(6) (2012), 1058–1069.

[5] Android Developers Documentation. Android 4.3 APIs: Restricted Profiles. http://developer.android.com/about/versions/android-4.3.html, last access 16.1.2014.

[6] Sony Corporation and Philips, Press release: Philips and Sony Announce Strategic Cooperation to Define Next Generation Near Field Radio-Frequency Communications, http://www.sony.net/SonyInfo/News/Press_Archive/200209/02-0905E/, last access 16.1.2014.

[7] J. Morak, H. Kumpusch, D. Hayn, M. Leitner, D. Scherr, F.M. Fruhwald, G. Schreier, Near field communication-based telemonitoring with integrated ECG recordings, *Applied Clinical Informatics* **2**(4) (2011), 481–498.

[8] G. Schreier, H. Eckmann, D. Hayn, K. Kreiner, P. Kastner, N. Lovell, Web versus app: Compliance of patients in a telehealth diabetes management programme using two different technologies, *Journal of Telemedicine and Telecare* **18**(8) (2012), 476–480.

[9] A.D. Lantada, C.G. Bris, P.L. Morgado, J.S. Maudes, Novel system for bite-force sensing and monitoring based on magnetic near field communication, *Sensors* **12**(9) (2012), 11544–11558.

[10] J. Morak, M. Schwarz, D. Hayn, G. Schreier, *Feasibility of mHealth and Near Field Communication Technology Based Medication Adherence Monitoring*, in: Proceedings of the 2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE, Piscataway, 2012. pp. 272–275.

[11] The Near Field Communication Forum, http://nfc-forum.org, last access 16.1.2014.

[12] NTAG203 – NFC Forum Type 2 Compliant IC with 144 Bytes User Memory, Product Short Data Sheet, http://www.nxp.com/documents/short_data_sheet/NTAG203_SDS.pdf, last access 16.1.2014.

[13] E. Chin, A. Porter Felt, K. Greenwood, D. Wagner, *Analyzing Inter-Application Communication in Android*, in: A.K. Agrawala, M.D. Corner, D. Wetherall (eds.) Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services (MobiSys'11). ACM, New York, 2011. pp. 239–252.