Design and implementation smart device control system based on indoor positioning

Doohwan Lim, Gilsu Lim, Jonguk Lee,
Minsu Kang and Sangjun Lee*

School of Computing,
Soongsil University,
Seoul, 06978, Korea
Email: oneof.neo@gmail.com
Email: jbkms137@gmail.com
Email: lju920@naver.com
Email: roroeiro@ssu.ac.kr
Email: sangjun@ssu.ac.kr
*Corresponding author

Abstract: With the rise of smart phone usage, smart phone users are the cause of many problems. They talk loudly in public spaces, divulging private information and taking photos in secure areas where cameras are not permitted. In this paper, we design and implement a smart phone control system to resolve such problems. The proposed system is based on an indoor positioning system and controls smart phone functions automatically by setting up restricted functions for the distinct zones.

Keywords: smart device; control; indoor positioning; MDM; mobile device management.

Reference to this paper should be made as follows: Lim, D., Lim, G., Lee, J. Kang, M. and Lee, S. (2016) 'Design and implementation smart device control system based on indoor positioning', *Int. J. Advanced Media and Communication*, Vol. 6, No. 1, pp.73–85.

Biographical notes: Doohwan Lim received his BS in School of Computing from Soongsil University, Seoul, Korea in 2015. His current research interests include mobile systems, information security.

Gilsu Lim received his BS in School of Computing from Soongsil University, Seoul, Korea in 2015. His current research interests include mobile systems and information security.

Jonguk Lee received his BS in School of Computing from Soongsil University, Seoul, Korea in 2015. His current research interests include web services and information security.

Minsu Kang received his MS and BS in School of Computing from Soongsil University, Seoul, Korea in 2014 and 2016, respectively. His current research interests include database systems and cloud computing.

Sangjun Lee received his PhD in the School of Electrical Engineering and Computer Science, Seoul National University, Seoul, Korea in 2004. He is

currently an Associate Professor of the School of Computing, Soongsil University, Seoul, Korea. His current research interests include database systems, mobile systems and cloud computing.

This paper is a revised and expanded version of a paper entitled 'Simple and flexible device control system based on indoor positioning' presented at *2015 International Conference on Platform Technology and Service*, Jeju, Korea, 26–28 January, 2015.

1 Introduction

Due to the rapid rise of smart phone usage in recent years, modern life has developed a close relationship with smart phones. However, there are many problems stemming from smart phone usage. Many concerns have been raised, such as making noise in public areas, infringing on intellectual property and leaking confidential information using cameras and recording devices.

To solve such security-related problems, this paper proposes a system that renders devices unusable in certain areas. Because the global positioning system (GPS) is difficult to estimate the precise location of indoor, many researches related indoor positioning technology has been conducted such as Sheinker and Ginzburg (2016) and Han et al. (2014). Therefore, our proposed system is implemented by combining indoor positioning technology and the mobile device management (MDM) system used on existing businesses such as Rhee and Na (2012). Furthermore, the system applies smart phone control for the MDM, beacon node-based 3D positioning recognition, such as Lee and Lee (2013) and a Kalman filter in Kalman (1960) to increase the accuracy of the positioning.

The remainder of this paper is organised as follows. Section 2 explains related exiting studies. Section 3 describes the proposed system. Finally, Section 4 presents the conclusion.

2 Related work

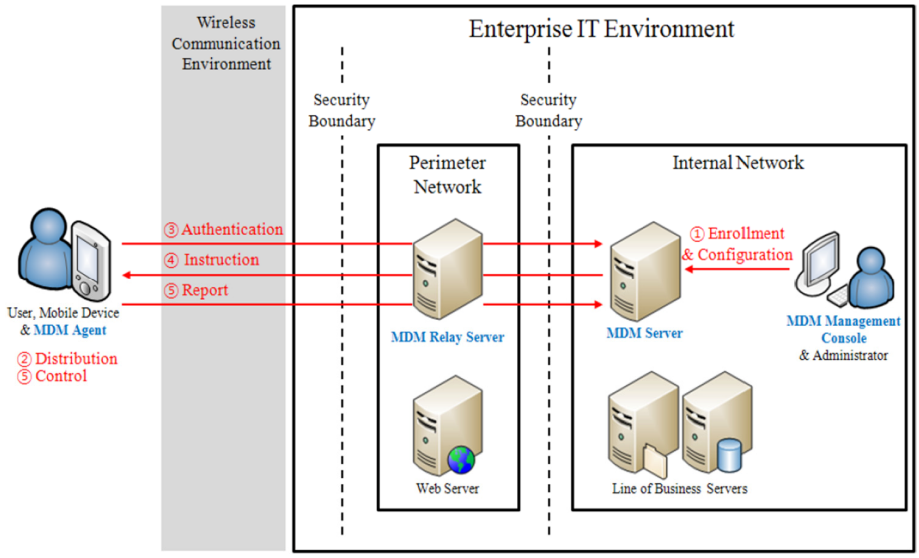
2.1 Mobile device management

Recently, growing interest in MDM system because of the increased need for mobile security as a result of bring your own device (BYOD) models that allow personal devices to be used in businesses. Miller et al. (2012) warned about the risks of BYOD as well as the rewards. The MDM system monitors the status of mobile devices through wireless connections and provides comprehensive management functions for controlling such devices depending on the situation, such as Rhee et al. (2012). In Figure 1, the general structure for MDM is depicted.

The MDM administrator is connected to the management terminal in order to manage the system and the control policies related to the monitored devices. The MDM client app that is installed on the smart phone manages the device, collects information and controls the smart phone in accordance with requests from the server. For security purposes, an internal network is used as the server. The server integrates the management of various

pieces of information registered in the system and provides appropriate services to the administrator and clients.

Figure 1 Overall MDM system schematic diagram (see online version for colours)



As shown in Table 1, apps of iOS cannot control a smart device. In this paper, we focused the management of smart devices based on Android.

Table 1 Scope of control for Android and iOS

Function	*Android*	*iOS*
Blocking recorder	O	X
Checking and controlling Wi-Fi	O	X
Controlling Bluetooth	O	X
SSID of wireless network	O	X
Controlling tethering	O	X
Blocking transmitting USB data	O	X
Checking state of USIM	O	X

2.2 Beacon node-based 3D positioning system

As shown in Figure 2, this proposed system calculates the coordinates of a moving node by measuring the distance between the moving node and three beacon nodes whose coordinates are already determined. The relative position of the smart phone can be calculated by measuring the distance between the smart phone and beacons that are installed on the ceiling.

Three spheres are calculated with radii equal to the distance between the moving node M and the centre of each of beacon A, B, and C installed on the ceiling, as shown

in Figure 3. By solving these three spheres simultaneously, two points can be derived, one above and one below the ceiling. The latter point indicates the location of the moving node.

Figure 2 Arrangement of nodes in 3D space (see online version for colours)

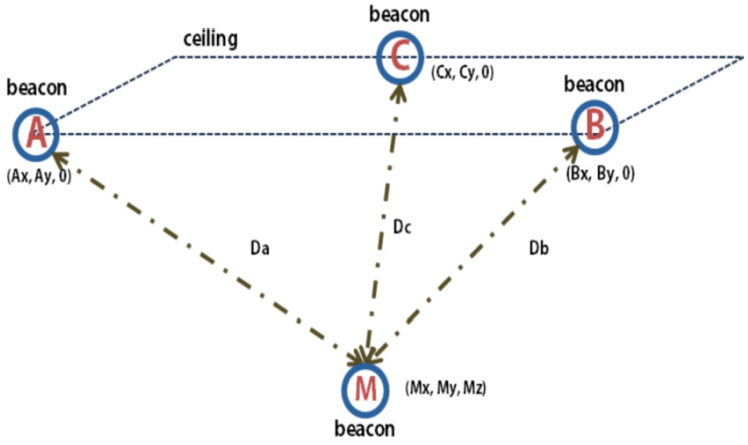
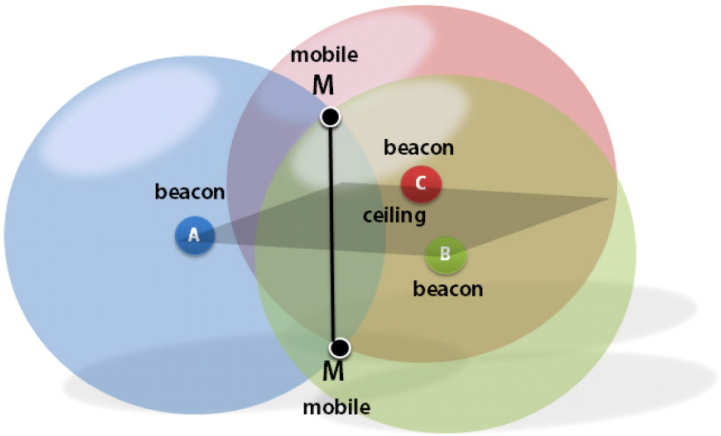


Figure 3 Schematic diagram of the three overlapping spheres (see online version for colours)



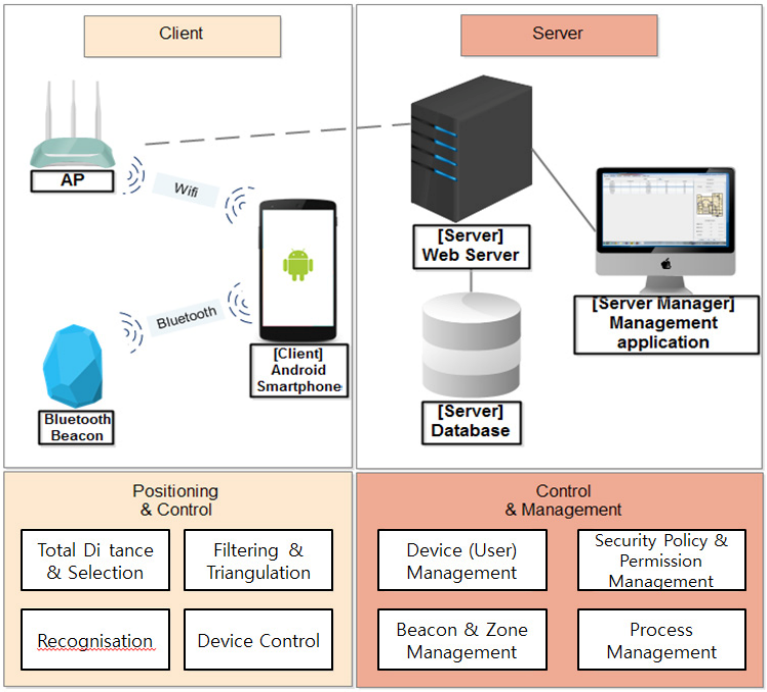
3 Proposed system

3.1 System overview

The system proposed in this paper consists of four main parts: the management program, the Android app, a Bluetooth beacon and the server as shown in Figure 4. The management program contains the security policy and stipulates the region where in the policy is applied. The Android app measures the distance using information from the user's smart phone and the received signal strength indication (RSSI) value from the pre-installed Bluetooth beacon such as Estimote Beacon (2015). The app then transfers this information to the server and controls the smart phone in accordance with the policy

received from the server. The server includes a database that stores the security policy and provides security-control services to the management program. The server also calculates the location of a user's smart phone and, depending on this location, issues the corresponding control policies.

Figure 4 System configuration (see online version for colours)



3.2 *Applying the technology*

Receive signal strength indicator

The RSSI value is used to measure the distance. Because the intensity of an electric wave attenuates as it moves farther away, the position can be determined by comparing the intensity of electric signals as shown in Figure 5. The measurement is conducted at the receiving end and it is based on statistical methods and a probability distribution.

Kalman filter

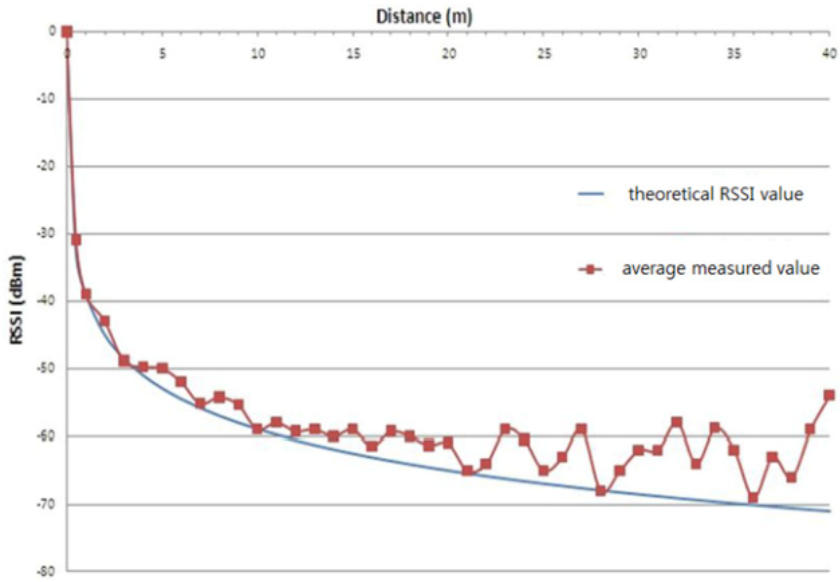
A Kalman filter is an algorithm used to improve the precision of an RSSI value, because these values are prone to interference and errors. The algorithm filters noise that is generated when signals are measured and it can accurately discriminate between signals. Applying this filter prevents the device from appearing to leap from one position to another, and it provides accurate and rapid positioning information for reliable detection.

Android JNI

Android supports application programming interfaces (APIs) and the java native interface (JNI) as development environments to design apps. The JNI framework can run libraries

written in languages other than Java, such as C or C++, as well as basic applications in java virtual machines (JVMs). JNI was used to apply the Kalman filter, and the proposed system controls devices using Android APIs and JNI in Android JNI (2015).

Figure 5 RSSI value for determining distance indoors in Benkic et al. (2008) (see online version for colours)



3.3 System structure

Smart phone client app

The smart phone client app, called user device inhibitor (UDI), is an app that provides convenience to users and security for administrators by automatically limiting functions in certain locations where smart phones are prohibited. Smart phone location information is identified using beacons, and the app consists of a login module, a positioning module, the device/app control module and a communications module. The login module for UDI establishes a connection to the server using either a telephone number or the device number. Once a user is registered in the system, subsequent logins are automatic.

UDI measures distance in real-time via Bluetooth communication with the beacons and sends the measurements to the server. The server can then determine the current location of the user and institute the control policy. UDI controls the smart phone stepwise according to the received control policy. User rights are managed through the positioning module, which is divided into function (or device) control and app control. Function control refers to the operations of the smart phone itself. Devices such as the camera and microphone are controlled through pre-emptive resource allocation, whereas other functions are controlled using APIs provided by Android. App control refers to the limitations stipulated in advance by the administrator for particular zones. Moreover, users can themselves limits specific apps on their phones by way of an allow/deny setting. As the Android platform becomes more sophisticated, it becomes more difficult to guarantee security when accessing apps or services. Because it is difficult to

completely limit the use of apps and services, apps that are prohibited are monitored by the service in the proposed system by introducing a screen when the corresponding app attempts to run, effectively blocking the user from running it. This function is implemented and operated in immortal mode.

UDI communicates with the server using the javaScript object notation (JSON) with the POST request method via an HTTP protocol. This way, the server receives the smart phone's device information, any personal information provided by the user, the beacon information and details regarding apps that are installed on the smart phone. To prevent binary files such as images from being corrupted during file transfers, encoding and decoding based on Apache Base64 is used.

When UDI is executed, it uses the login module to verify whether the device is registered. If it is not, registration is needed before the device can connect to the server. After successfully logging in, detailed information including user's rights is registered in the server, which then locates the device's position. A value is calculated with the positioning module, and this is used to locate the device. The server transfers functions and app-control policies set by the administrator in advance from the secured zone to the UDI app. These policies are set to the corresponding zone and used to control the smart phone. This app sends and receives encoding data with Base64 after processing it in the JSON format. It operates in request–reply mode, because its communication protocol is based on HTTP. Figure 6 shows the control flow for the proposed app.

Figure 6 Smart phone app-control flowchart (see online version for colours)

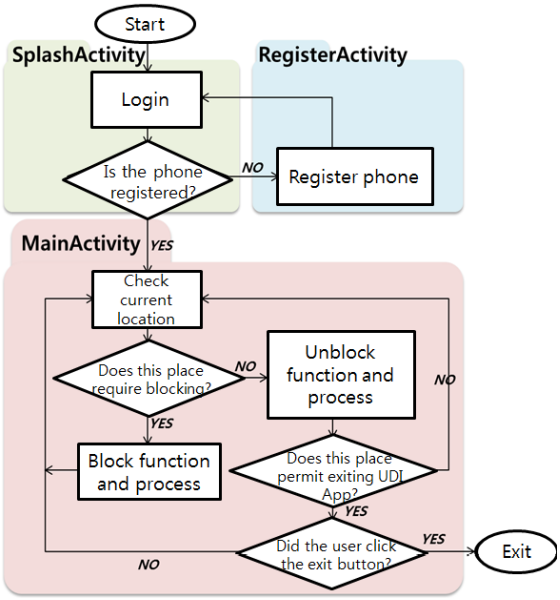


Figure 7 shows a real execution screen of the android app implemented in this system. In Figure 7(a), the app attempts to login automatically using the device information and the phone number from the smart phone. If the login is unsuccessful, a registration process follows, as shown in Figure 7(b). If successful, a personal monitoring screen appears in Figure 7(c). When the blocked-apps list is selected, a list of restricted apps installed on the smart phone is shown, as seen in Figure 7(d).

Server management application

The system management program proposed in this paper includes all of the resources for the system, such as registered devices, beacons, maps, security zones and new creation/modification/deletion/search functions for the apps through the program shown in Figure 8. This program is used to control and coordinate the overall system including user logins and logouts, zone arrivals or departures, a messaging function and a user-monitoring function to manage registration requests, etc. With this program, the administrator can effectively control smart phone usage commensurate with the security policies in the corresponding zone, and security-related abnormalities that occur in this zone can be monitored.

Figure 7 Running images on Android App: (a) initial connection screen; (b) member (terminal) registration; (c) function control screen and (d) App control screen (see online version for colours)

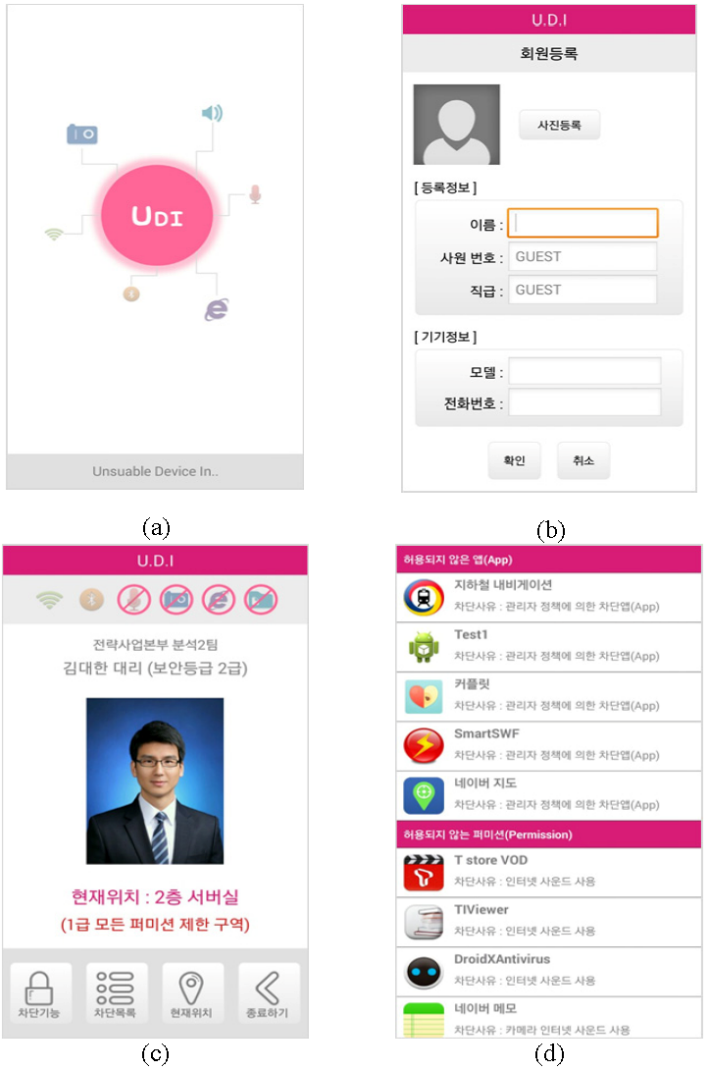
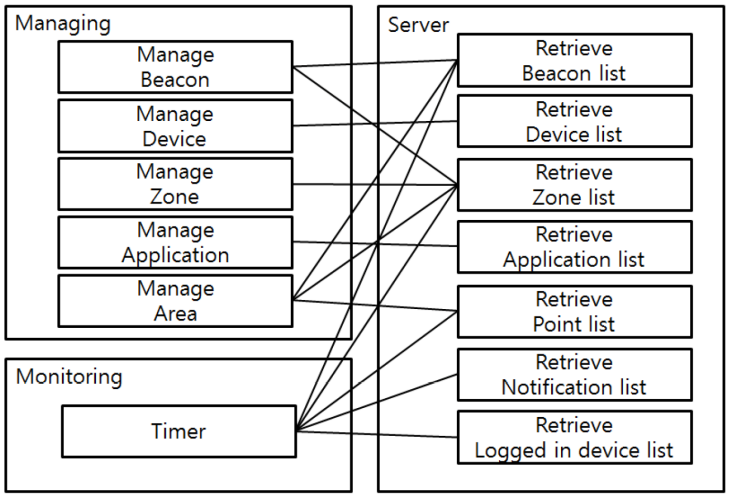


Figure 8 Configuring the system management functions



The management program consists of three modules: a management module, a zone-monitoring module and a communication module. The management module is divided into five submodules: device management, map management, beacon management, security-zone management and app management. The zone-monitoring module comprises message-log monitoring and device monitoring within the zone. The communication module comprises the HTTP communication sub-module and a UDP socket-communication sub-module.

The management module is used to access various information needed to operate the system and to manage them according to the security policy.

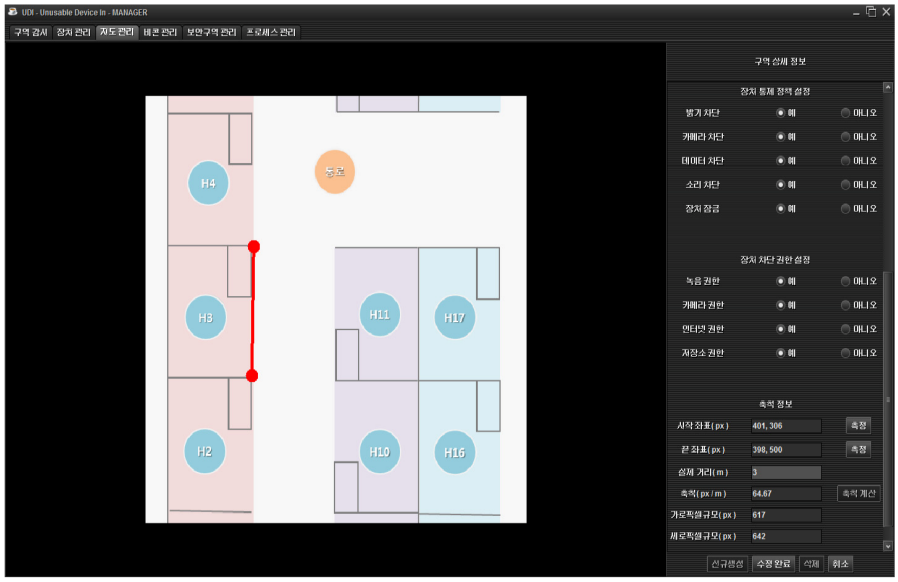
The device-management module is used to manage information and to identify the smart devices introduced into the security zone, along with information about the owners of the devices. This module is designed to determine the respective users for each device.

The map-management module is used to register 2D maps for each of the security zones. As shown in Figure 9, an interface is available for administrators, with which security zones can be designed and security policies can be defined and applied to these zones after the scale has been determined. When two points are provided in designing the zone, the distance between them is calculated. The ratio (pixel/m) of the pixelated distance between two points to the actual distance in the area is calculated. The scaling ratio and the screen size at are stored together at this time so that the map of the zone on the monitor can be adjusted depending on the screen resolution.

To maximise details in the map, the canvass area where the actual map is drawn includes a margin of 10% of the screen along the periphery of the map. In addition, to minimise unnecessary white space and to construct a figure in accordance with the canvass size, the numbers of x and y pixels on the canvass are divided by the number of x and y pixels in the original map image. This way, the magnification ratio for the horizontal and vertical sides of the canvass can be calculated.

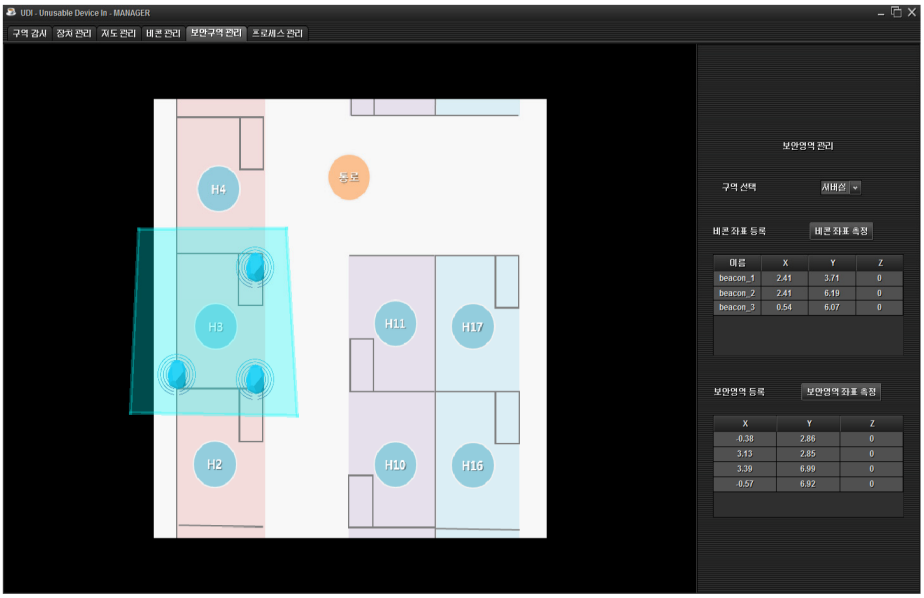
The beacon-management module is designed to manage registration and identification information for the beacons used to measure the users' positions. The beacons are registered in each security zone, so that devices going in and out of these zones can be detected.

Figure 9 Map management: inputting security policies and scale information (see online version for colours)



The security-zone management module is used to set up coordinates for the registered beacons on the registered maps, and to register the security zones. As shown in Figure 10, when clicking on the map, the coordinates are calculated and registered according to scale. The system was designed to control devices according to the control policy once a device is detected in a security zone.

Figure 10 Security-zone management: beacon and secure-area coordinate setup (see online version for colours)



The app-management module is used to show that apps that are running apps on the user's smart device, and to assign control policies for each app by the administrator. This module was designed to always allow or block apps depending on whether they are judged trustworthy or harmful.

The zone-monitoring module monitors which devices are entering or leaving a zone when the system is running and responding appropriately.

The message-log monitoring module displays log records for major events that occur in the security zone received by the server and displayed on the screen. With this module, an administrator can recognise what kinds of events are currently occurring in the system.

The device-monitoring module displays the number of devices used in the security zone and their coordinate information. An administrator receives detailed information regarding the devices in the current security zone.

The communication module is needed for exchanging control messages and data between the management program and the server. Communication with the management module is done via the HTTP POST request method, and data are exchanged using the JSON format with parameter parsing. A function for fetching system logs from the monitoring module is provided by UDP sockets, and multicast is used to deliver messages from the server to a number of management programs.

When the management menu is selected, data stored in the current management server is displayed. An administrator can view the full list or selected portions of it by searching with specific conditions, and data can be added, updated and deleted if necessary. When an administrator requests a required function from the server through the management program, the server performs this request and returns the outcome.

A login process is required when the management program connects to the server. When the login is successful, it establishes HTTP communication with the server, and monitoring information is subsequently received from the server periodically. As shown in Figure 11, security zones in currently monitored areas can be confirmed, along with the beacon and user locations and login information is recorded in the server.

Server

The server is run with a web servlet. The servlet manages the database for the system and provides the services required by the Android client and the manager's application programs.

The positioning algorithm introduced in Section 2.2 performs poorly when two beacons appear in a straight line along the *X*-axis. In addition, its calculation method forcefully requires the first beacon's location as an original point. This means that it is not easy to install the beacons in arbitrary locations. Thus, a conversion process is performed to apply any arbitrarily located beacons to the positioning algorithm.

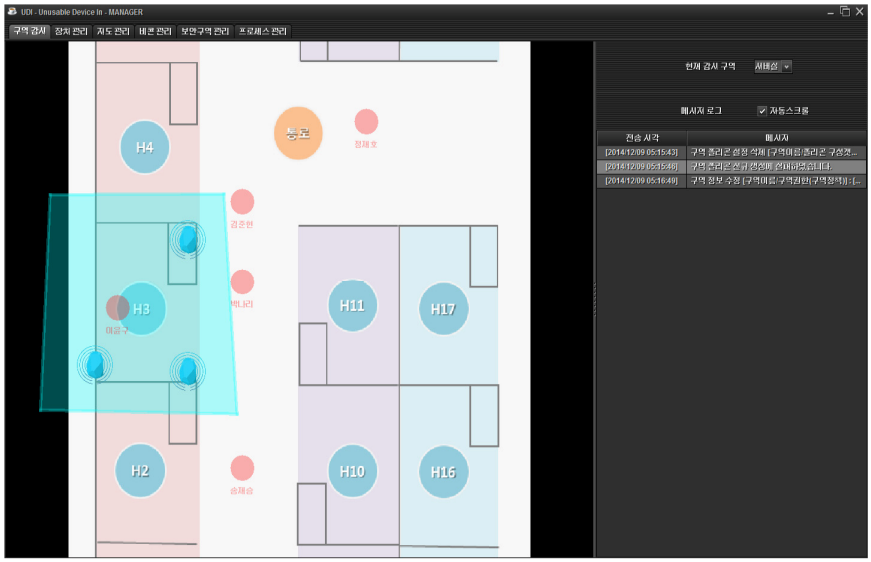
3.4 Operations of the overall system

A client app for UDI must be installed on the smart phone before the proposed system can operate. Furthermore, at least three beacons must be installed in any security zone. Once the beacon information is provided for the security zone and the control policies are in place, the system can be initiated.

When a smart phone enters a security zone, the beacons detect it automatically and the distance between the beacon and the smart phone is measured and sent to the server.

Here, the Kalman filter is applied to compensate for errors that occur when measuring the distance with the RSSI value.

Figure 11 Zone monitoring: administrator notification function (see online version for colours)



The server calculates the location of the smart phone using the algorithm introduced in Section 2.2 based on the distance of the smart phone. The server then issues the appropriate control policy to the smart phone for the current security zone.

The smart phone is controlled according to the control policy received from the server. The smart phone's position is verified periodically, so that when users leave the security zone they can use their smart phones normally.

3.5 *Analysis on the proposed system*

Smart phones offer countless functions besides telephone calls and messages, such as personal schedule management and photography. When popular apps are restricted by the MDM service, users may feel that this is inconvenient, and they may consider using other devices to perform these functions. The proposed system controls smart phones only when they enter specific areas, and it minimises such inconveniences by implementing several functions, such as process control, smart phone function control, app deletion and measures to protect against lost smart phones. UDI implements a module for restricting functions in secure locations through indoor positioning, and it provides convenience to users in addition the above functions. An administrator can set up restricted zones by demarcating areas that require restrictions from those that do not. APC application program is used to apply these restrictions and to specify policies for particular smart phone functions in specific areas. Moreover, the security function can be further enhanced, because the location and the distribution of smart phones can be determined. Functional restrictions to the user's smart phone can be controlled for different areas with the application program, and these restrictions can be applied immediately. Thus, for added convenience, security functions can be strengthened without additional updates.

4 Conclusion

In this paper, a system was proposed for restricting smart phone functions in security areas using an indoor positioning system based on a triangular positioning method using beacons. The proposed system controls smart phone functions automatically by setting up restricted functions for the distinct zones, and it provides monitoring functions for managing the program. With the proposed system, issues pertaining to information leaks and excessive noise can be readily solved by providing precise restrictions in particular zones based on indoor positioning, rather than existing GPS-based static services. In addition, by restricting functions and security policies in real-time, we expect that the proposed system will be used in various environments where dynamic control of Android-based smart phones is required.

Acknowledgements

This research is supported by the MSIP (Ministry of Science, ICT and Future Planning), Korea, under the Seoul Accord Vitalization Program (IITP-2015-R0613-15-1175) supervised by the IITP (Institute for Information and communications Technology Promotion).

References

Android JNI (2015) http://developer.android.com/training/articles/perf-jni.html (Accessed 25 February, 2015).

Benkic, K., Malajner, M., Planinsic, P. and Cucej, Z. (2008) 'Using RSSI value for distance estimation in wireless sensor networks based on zigBee', *Proc. Of IWSSIP.2008: Systems, Signals and Image Processing*, Bratislava, Slovak Republic, pp.303–306.

Estimote Beacon (2015) http://estimote.com/ (Accessed 25 February, 2015).

Han, D., Jung, S., Lee, M. and Yoon, G. (2014) 'Building a practical Wi-Fi-based indoor navigation system', *Pervasive Computing IEEE*, Vol. 13, No. 2, pp.72–79.

Kalman, R.E. (1960) 'A new approach to linear filtering and prediction problems', *ASME-Journal of Basic Engineering*, Vol. D., No. 82, pp.35–45.

Lee, H. and Lee, D. (2013) 'The 3-dimensional localization system based on beacon expansion and coordinate-space disassembly', *Journal of The Korean Institute of Communications and Information Sciences*, Vol. 38B, No. 1, pp.80–86.

Miller, K.W., Voas, J. and Hurlburt, G.F. (2012) 'BYOD: security and privacy considerations', *IT Professional*, Vol. 14, No. 5. pp.53–55.

Rhee, K. and Na, H. (2012) 'Security test methodology for an agent of a mobile device management system', *International Journal of Security and Its Applications*, Vol. 6, No. 2, pp.137–142.

Rhee, K., Jeon, W. and Won, D. (2012) 'Security requirements of a mobile device management system', *International Journal of Security and its Applications*, Vol. 6, No. 2, pp.353–358.

Sheinker, A., Ginzburg, B., Salomonski, N., Frumkisc, L., Kaplanc, B.Z. and Moldwina, M.B. (2016) 'A method for indoor navigation based on magnetic beacons using smartphones and tablets', *Measurement* (*Journal of the International Measurement Confederation*), Vol. 81, pp.197–209.