IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 18, Issue 3, Ver. II (May-Jun. 2016), PP 89-95
www.iosrjournals.org

Design and Implementation of Location-Based Application of Secure Coding Providing Local Information

Yangwon Lim[1], Jinyoung Jung[2] , Hankyu Lim(corresponding author)[3]
[1](Auto Industrial co., Korea)
[,2,3](Multimedia Engineering Department, Andong National University, Korea)

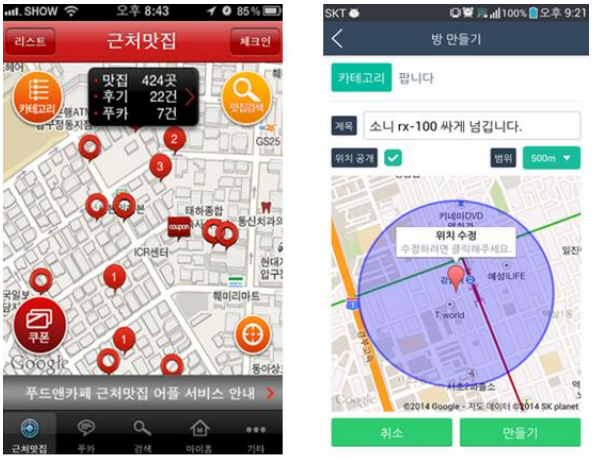
Abstract: *The development of information technologies (IT) has made it easier to produce and process information. In particular, the development of smartphones has enabled anyone to easily access and use processed information. These technologies are used not only online, but also by connecting with offline services. In recent years, services that provide real-time travel information, which are connected with offline services, have become popular. In addition, security should be maintained not only for information provided by the server, but also for the protection of user information. The Secure Coding technique is very important and it should be considered in developing the applications. This paper made use of a location-based service to design a new method of providing local information to users so they can view information more selectively by designating the size of the region from which the information comes, as well as the amount of such information.*
Keywords: *Location Based Service, Local Information, Smartphone App , Secure Coding.*

I. Introduction

THE Internet has evolved into a new paradigm based on Web 2.0. In the past, users were only allowed to view information on the Internet. However, Web 2.0 has enabled them to create information on their own and share it easily [1]. As users produce and share information together, the Internet environment has shifted toward the participation of users [2]. As users of smart devices have become more experienced, they can now install applications they want more easily than on PCs [3]. Moreover, as smart devices contain cameras, GPS technology, and sensors, such as acceleration sensors, diverse applications that make use of these tools are simultaneously developed and distributed. Among these applications, the growth in the use of living-oriented applications that provide information about food, tourism, traffic, etc. using location-based service (LBS) applications is especially remarkable [4]. In this paper, an LBS application is introduced that provides local information, including tourism, culture, traffic, etc., in real time based on location information about the user in motion. Differing from previous applications, the proposed LBS application provides users with only information that is relevant to their current location. Moreover, users are provided with only desired information by setting the information type and size (amount) themselves. Servers and client applications have been designed and implemented to protect their communications and resources from external attacks.

II. Existing App Using LBS

Most applications launched these days provide information in a region of a certain size. Contents are rearranged according to an individual's location. Therefore, those applications provide not only information on the contents themselves, but also location-based information including metadata for those contents [5]. Moreover, although they provide information based on location, they usually generate information regarding a single field. Therefore, users must install a number of applications to obtain the desired information. Figure 1 (left) below shows a food information application and it shows there exists a large amount of information in an area of a certain size. Figure 1 (right) shows a bartering service application for a used goods market. It has a disadvantage in that users who want diverse information about different regions must install multiple applications at the same time. Moreover, the size of the field serviced in a metropolitan area should be different from those of small and medium-sized cities. With the progression of recent studies on the relationship between location-based users, not only the user's current location and information on the respective region, but also the relationship between users in the respective region are emerging as important factors [6].



Food & Café Application **Multipurpose Chatting Application**

Fig. 1 Location based Application

A. Tour API

Tour API is an open API that provides users with tourism information and it is distributed by the Korea Tourism Organization [7]. Tour API supports nine languages, including Korean, English, Chinese, etc., and a user issued a developer key can request data from Tour API along with a certain type of URL to retrieve tourism data. There are two types of data provisions, i.e., XML and JSON, and the developer can make use of data through the parsing procedure [8]. It provides information about tourism, food, cultural facilities, accommodations, etc. and it was used when searching and analyzing information about tourism, food, and cultural facilities in this paper.

B. Secure Coding

Secure coding is a static analysis technique. Unlike dynamic analyses, it does not have software executed. It rather finds vulnerable areas from the viewpoint of security, including the use of logics lowering quality and vulnerable functions in source codes, and the absence of the verification of input values. While secure coding is applied to specific algorithms, it also includes a security mechanism for images themselves [9]. In recent years, security guidelines have been suggested to help develop safe codes against external attacks when designing and implementing mobile applications [10]. For software code that does not consider secure coding, the following methods are used to detect errors: path-sensitive analysis, flow-sensitive analysis, value analysis, inter-file analysis, field-sensitive analysis, context-sensitive analysis, pointer analysis, and loop invariants inference. The Ministry of Public Administration and Security in Korean Government classified vulnerability of the Android-Java programs into input data verification and representation, API exploit, security features, time and status, error handling, and code quality and encapsulation. Security features can cause many problems if the use of them is not considered very carefully. The following items should be considered in security features: authentication, access control, confidentiality, encryption, and rights management [11].

In particular, when mobile applications are developed, weak points that can emerge due to the use of vulnerable APIs should be minimized. In addition, as mobile applications are the systems in which server communications frequently occur, they require the proper handling of vulnerabilities in the verification of input data, security functions, and encapsulation.

III. Design Of Local Information Using Lbs

The LBS application suggested in this paper to provide local information was designed with eight category menus, including local community, comprehensive local information, tourism information, local food, recruitment, amenities, cultural facilities, and weather, where the user can select the information they want to view. The menu and setting structures are defined in Figure 2 below, so the location and the amount of information can be controlled.

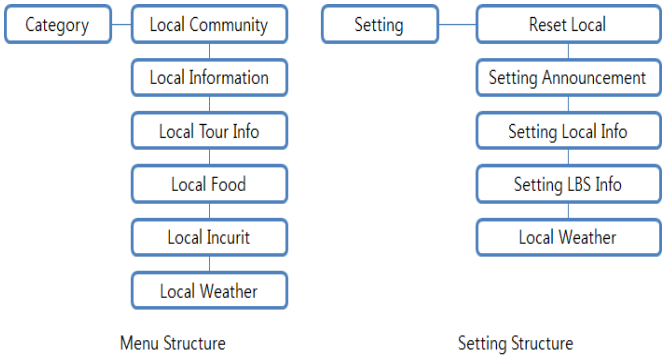


Fig. 2 Location based Service Menu and Setup Menu

When the application is first launched, it provides only the previously set information about the current location by retrieving the current location's GPS coordinates. However, if the user wants to learn about information in another region beforehand, it is designed so he/she can select the region in the settings. Figure 3 and 4 below shows the UI, where a user chooses local information to view information about a certain region.

As previous location-based applications show information within a certain distance from the user's current location, large amounts of information are generated in metropolitan areas, and it is possible the application cannot find information in small and medium-sized cities. The application suggested in this paper designates the amount of displayed information and shows only 10 pieces (default value) of information that are most relevant to the current location. Moreover, for more information, the application allows the user to set the amount of displayed information so he/she can search the desired information more easily without becoming confused.

Fig. 3 UI : Selecting Local Information – Select Area(1)

Fig. 4 UI : Selecting Local Information - Select Area(2)

Figure 5 below shows a module structure that locates the user and provides information about the relevant location. The analysis module has a function of recalling only the required information from storage to send to users by capturing the information about the relevant location and checking the amount of generated information

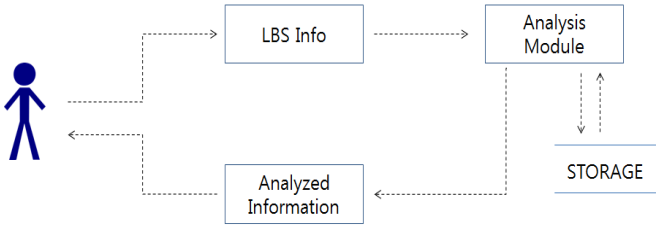


Fig. 5 Module Structure

As this shows, the user-to-user module was designed to consist of the server that stores resources and the application that makes the user's access more convenient. Figure 6 is the basic conceptual map that exhibits the server and the sharing of information among users. A user can share local information and his/her current location with the other users in similar locations to his/her current location.

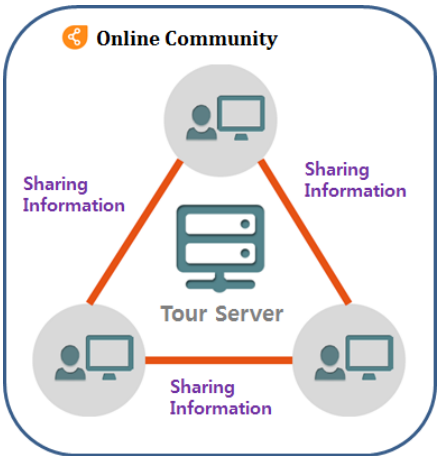


Fig. 6 Basic Diagram of User Community

IV. Implementation

This study proposed a location-based regional travel information system. Its server was configured based on PHP and MySQL. The client application was configured using HTML5 in the WebView format. Figure 7 shows the architecture of the location-based regional travel information system that consists of the server and the client. The server was configured to provide data in the Web service format to facilitate access to it for management. The map data, which generate the network's heavy traffic due to massive amounts of information, were designed to enable the transmission of information using only metadata for simultaneous processing on the server and the client. In addition, the server was designed to be operable through the separate configuration of MAP, TOUR, Server PUSH, and CHAT API to process huge volumes of resources by phase. The client application was processed to enable the retrieval of the server's information using open APIs such as Tour API and MAP API via a GPS module. The stored addresses, local codes, user information, local information, etc. were stored in privatized storage. The application was implemented to provide information regarding a user's current location and region using this storage information. Moreover, requests for resources were minimized by requesting data based on server and URL/JSON type. The majority of information was stored in privatized storage to ensure a prompt response to the user's request is possible. In doing so, the level of security can be increased through minimal data processing between the server and the client. Moreover, within the server, individualized storage areas, resources, and APIs were separately designed, and thus the security of personal information and system resources could be maintained.

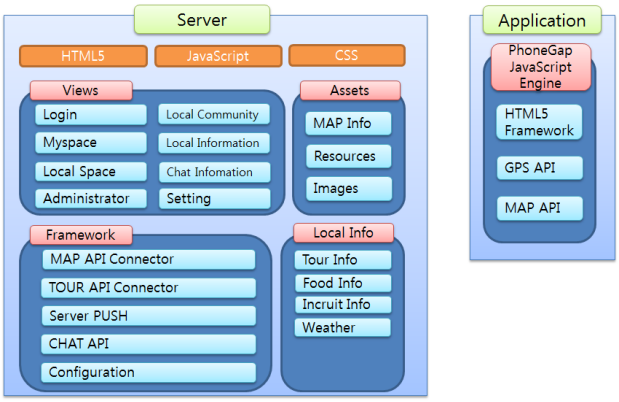


Fig. 7 Architecture of Location based Local Tourist Information System

C. Local Information Update Method

The user's location information was divided into automated location information based on the current location and virtual location information according to the user's own setting. As mentioned earlier, this refers to not only information on the current location, but also virtual location information that allows the retrieval of information on the other locations. It was composed of a module that revises the user's location, which is implemented when the location revision icon of the action bar on the main screen is touched after initial launch. It retrieves a coordinate by selecting the optimal provider from methods using GPS or base station according to the device settings. The relevant coordinate is converted to an address using the MAP API service and the address is subsequently provided to the user, followed by a series of storing processes that keep coordinate values, addresses, and local codes necessary for other services inside the app. The user registration process that pops up at the initial running stores the ID and nickname of the Android device in the server. Even after reinstalling after deletion, it remembers the previous nickname using the Android device's ID. Figure 8 below shows the location setting execution screen.

Fig. 8 Location Setup Screen

When the user information is renewed, an alarm service is provided that enables the user to reset his/her location by agreeing to activation of the service. This alarm service can be set in the environment setting menu. In addition, as shown in Figure 9, a user can receive detailed information on a specific location using virtual location information according to the size set by the user. The user can also utilize information on the respective region, and its travel distance and local community in the virtual location.

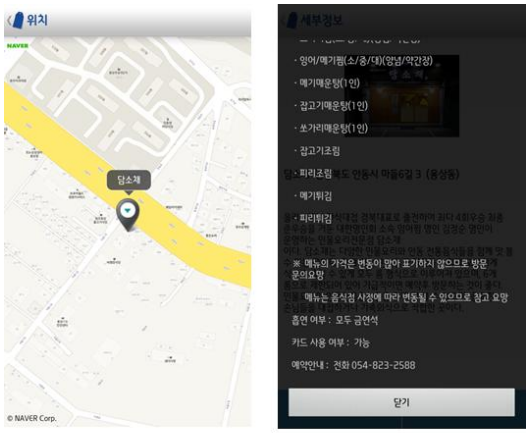


Fig. 9 Local Information providing Virtual Location

D. Tour API Module

This module provides information about local food, tourism, and cultural facilities to users of Tour API. When selecting a function, each menu completes different URLs to request of Tour API, so requests regarding food, tourism, and culture can be processed on a single module. The processing of each URL in this manner helps operate system resources and personal information separately, thereby maintaining the security of system resources. Tour API is available in two types: XML and JSON. Because the JSON type has a comparative advantage in speed in the case of small-sized data processing, data processing was designed and implemented following JSON in this paper. The JSON object was objectified through the parsing process and stored in an array list to be used. Figure 10 below shows the screen where the information is being provided using Tour API based on the current location. The location information set by the user is used to provide a list of restaurants including information and images starting from the nearest restaurant. Clicking on "detailed view" delivers information about a particular restaurant and its accurate location.

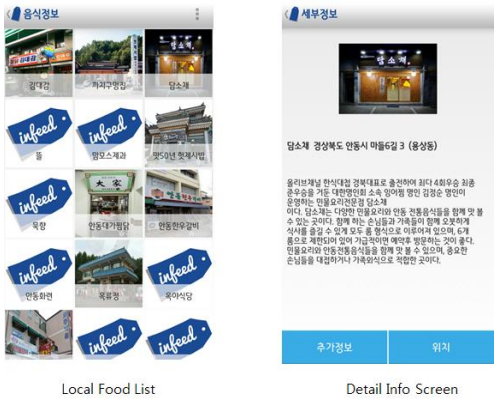


Fig. 10 Local Information

E. Secure Coding for Location-Based Application

The server and the client application are required to maintain different forms of security codes to protect information between the server and the application. This study configured security codes for the security of user information, communications between system resources, and contents within the application. Figure 11 presents security codes for the application that provides local information.

```
.....
.....
JSONObject jsonObject = new getJSON().getlocationInfo(params[0], 0);
    try {
            jsonObject = jsonObject.getJSONObject("result");

            check = jsonObject.getBoolean("code");
            if (!check)
                msg = jsonObject.getString("codeMsg");
            else{
                msg=jsonObject.getJSONArray("data").getJSONObject(0).getString("Nick_Name");
            }
    }
    catch (JSONException e) {
            e.printStackTrace();
    }
    return check;
.....
.....
```

Fig. 11 Security Codes for Local Information

This study was designed and implemented based on the verifications of input data, which cover the omission of the verification of program input values or inappropriate verifications, and the incorrect designation of data formats. The study also intended to include proper security functions for the control of access to the server, encoding, and the management of information rights. Moreover, it aimed to minimize the user's data leakage by encapsulating important data among the resources operated by the server or independent functionality.

V. Conclusion

LBS applications should be able to change adaptively according to the surrounding environment of the user's current location. However, previous LBS applications were inconvenient, as they were unable to provide information by designating a certain category or a certain distance by retrieving GPS coordinates, such as several meters away. The service application suggested in this paper was designed and implemented by supplementing these shortcomings and made it possible for users to choose selectively what information they want to receive by category so users can be provided with information more accurately and conveniently. In addition, the study designed and implemented the system that can protect resources from external attacks through the control of access and the encapsulation of server resources and APIs for the security of the mobile application in which server communications frequently occur. To respond to technological developments and users' tastes, LBS applications should be able to show intellectually only information users want to see. With its active location setting and information provision abilities, we expect the LBS application suggested in this paper can contribute to the development and offering of diverse applications with decent usability in the future. Secure coding is essential for developing these applications. The developer should follow secure coding guidelines and examples of violations. Recent social network services (SNSs) also offer mobile services, rather than personal computer (PC) services, that can provide location-based services as a priority. In the future, these mobile services are likely to provide location-based services that use the user's present location as well as the user's scope of movement. Based on the results of this study, a future study will focus on the encapsulation of modules and the control of access for the security of location-based information services through the analysis of location information about the user's scope of movement and pattern analysis of other users' movements in similar locations.

Acknowledgements

This work was supported by a grant from 2016 Research Funds of Andong National University.
A corresponding author is professor Hankyu Lim(hklim@andong.ac.kr).

References

[1] O'REILLY, T., "What Is Web2.0 : Design Patterns and Business Models for the Next Generation of Software", COMMUNICATIONS and STRATEGIES vol.65, 1st quarter 2007.
[2] MITSUTERU NAKAMURA, YASUHIRO OMIYA, KAZUKI KATAGISHI , "A Secure Coding for Function-Approximated Images", 12th WSEAS International Conference on COMMUNICATIONS, Heraklion, Greece, July 23-25, pp.416-420, 2008
[3] http://blogs.wsj.com/korearealtime/2014/12/12/smartphone-usage-overtakes-pcs-in-south-korea/?mod=WSJBlog&mod=koreart
[4] http://play.google.com/
[5] Yangwon Lim, JunYeong Kim, Daejun Lim, SeungJae Lee, Hankyu Lim, "A Study of Social Network Content for Effective Content Displays-Centering on Travel Information Content-", ICFICE 2014,
[6] Chun-Der Chen, Rouh-Yun Yu, Chia-Huei Ho, Chin-Tsai Lin, "Customer Value Influences on Relationship Quality and Subsequent Behavioral Intentions: An Empirical Study of Location-Based Service Platforms", Mathematics and Computers in Contemporary Science, pp.78-83, 2013.
[7] http://api.visitkorea.or.kr/
[8] Crockford, Douglas. "The application/json media type for javascript object notation (json)." (2006).
[9] Nakamura, M., Omiya, Y., Katagishi, K., Morooka, Y., Toraichi, K., & Murakami, H. (2008, July). A secure coding for function-approximated images. In WSEAS International Conference. Proceedings. Mathematics and Computers in Science and Engineering (No. 12). World Scientific and Engineering Academy and Society, 2008.
[10] Hopkins, Paul. "Secure Systems Development and Secure Coding."Engineering & Technology Reference 1.1 (2014).
[11] Hyeonpyo Hong, Hankyu Lim, Application of Smart Device Usage Control adopting Secure Coding for Hacking Protection, International Journal of Internet and Web Application, 2015