Information gathering system based on BLE communication for bus information sharing

Katsuhiro Naito
Department of Information Science, Aichi Institute of Technology,
1247 Yachigusa, Yakusa, Toyota, Aichi 470-0392, Japan

Katsuyuki Tanaka
Graduate School of BusinessAdministration and Computer Science,
Aichi Institute of Technology,
1-38-1 Higashiyamadori, Chikusa, Nagoya, Aichi 464-0807, Japan

ABSTRACT

Information gathering system is focused to manage service status in public transportation. Smartphones are first candidate devices for this purpose because they have an internet connection through a cellular network. Participatory sensing methods, where many common people collaborate to collect information on their own smartphone, have been focused on information gathering. A sensing system requests participants to collect demanded information. On the contrary, they may not be enough to collect information continuously because ordinary people may not exist in a service area.

This paper proposes a collaborative mechanism with BLE devices with sensors and smartphones to realize the information gathering system. The proposed mechanism employs Bluetooth Low Energy (BLE) devices as a beacon device that triggers our special application for smartphones. BLE devices also store demanded information with sensors. The special application can be automatically launched when a beacon message arrives on the smartphone and can obtain the demanded information from the BLE device. The benefit of the proposed system is a hands-free operation of the smartphone application because the information gathering process is performed in a background process. Experimental results demonstrate that the developed application can measure its position with a GPS receiver and can upload the obtained location to the cloud automatically.

Keywords: Information gathering system, Bus location system, BLE, Participatory sensing

1. Introduction

Information gathering system in transportation systems is becoming popular to enhance the effective operation and service quality of public transportation[1], [2], [3]. Especially, a realtime location and a number of passengers are important information for passengers. The traditional systems employ a special device with a network module for cellular network modules, a special mesh network, etc. to collect various information such as a location, a number of passengers, an operational status, etc. to enhance service quality[4]. Therefore, the cost of the special device is a big issue for installation and long-term maintenance.

Some researchers have tried to use a consumer device to collect information as a new information gathering system[5]. Employing smartphone sensing is a new trend to realize an information gathering system in public transportation vehicles[6]. Since the typical cost of consumer devices is less than the special devices, consumer devices are easy to install and to launch a new system. In addition, maintenance of consumer devices is also easier as a replacement and a repair compared to the special devices.

Recently, some researchers have tried to realize participatory sensing, where many common people collaborate to collect information with their own smartphone[7], [8]. Therefore, bus location service using participatory sensing has been proposed[9], [10]. Participatory sensing is a flexible way to collect a bus location and a number of passengers on a bus. On the contrary, participants should collect demanded information and upload it to a system. Therefore, the system requires a lot of volunteers for uploading the demanded information in the practical consumer service. Additionally, recent transportation service requires various kinds of information such as operational status, mechanical status, etc. However, this information is difficult to obtain by passengers' smartphones. As a result, typical participatory sensing methods may not be suitable for general consumer services including the public transportation service.

This paper proposes a collaborative mechanism with Bluetooth Low Energy (BLE) devices with sensors and smartphones to realize the information gathering system. The features of the proposed system are flexibility for collecting various information by a microcomputer board and reasonable cost for installation and operation by smartphones. Additionally, an automatical collection mechanism is suitable for practical consumer services. The proposed mechanism employs BLE devices as a beacon device that triggers our special application for smartphones. Additionally, BLE devices also store demanded information from sensors. The special application can be automatically launched when a beacon message arrives

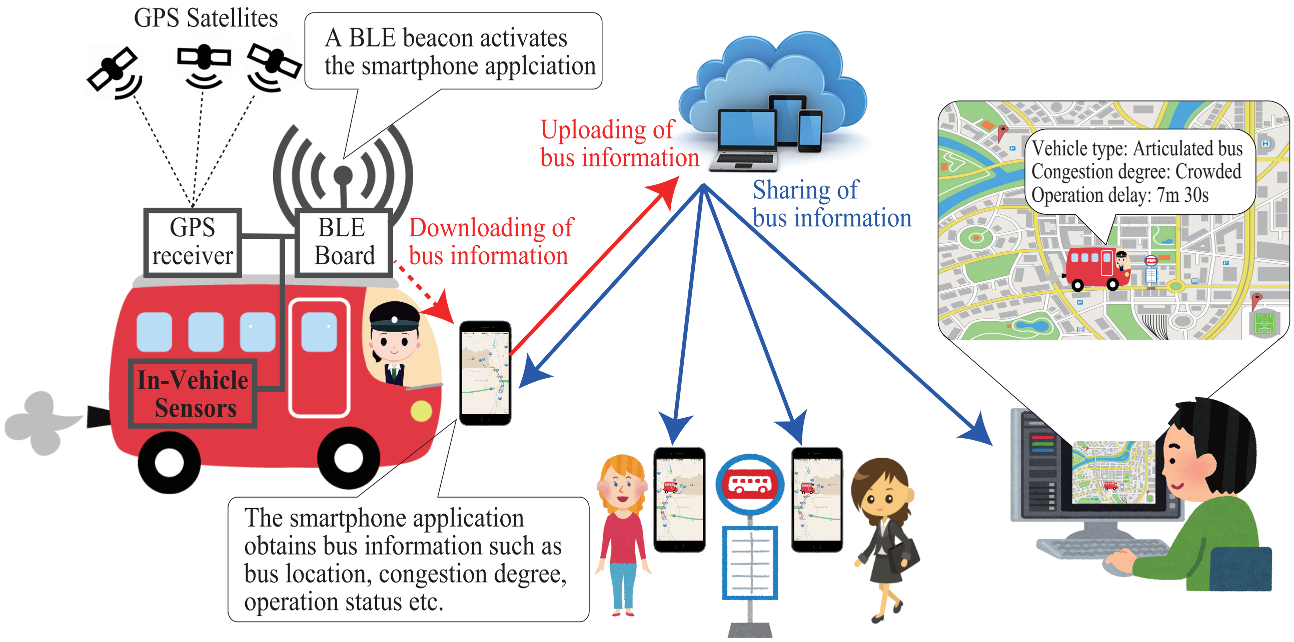

Fig. 1. Cooperative information gathering system with a BLE device and smartphones.

on the smartphone and can obtain the demanded information from the BLE device. The contributions of this paper are 1) proposal for collaborative mechanisms of the beacon mode and the communication mode in BLE, 2) proposal of the BLE based automatic information gathering system, and 3) development of the prototype implementation.

2. Bluetooth Low Energy

BLE is a wireless personal area network technology designed by the Bluetooth Special Interest Group (SIG). It uses the 2.4 GHz radio frequencies. On the contrary, it uses a different set of channels. BLE uses 40 channels instead of 79 channels in Bluetooth 3.0 and reserves three channels for an advertisement. The advertisement channels are randomly selected to reduce interference. The discovery latency depends on the three parameters: the advertising interval, the scan interval, and the scan window. BLE supports two types of communication: one-directional beacon broadcasting and bi-directional communication.

2.1. Generic Attribute Profile

The system model of BLE consists of a peripheral and a central. The peripheral broadcasts an advertisement message periodically on an advertising channel. Bluetooth 4.0 introduces Generic Attribute Profile (GATT) instead of the application profiles defined in Bluetooth 3.0. GATT provides common operations and a framework for data transportation by the attribute protocol. Attributes are formatted as services and characteristics.

Fig. 2 shows the structure of GATT. Services may contain a collection of characteristics. Characteristics contain a single value and any number of descriptors describing in the characteristic value. The central can access the characteristic values by the read and write methods. As a result, GATT provides a flexible data communication scheme according to a target service for developers.

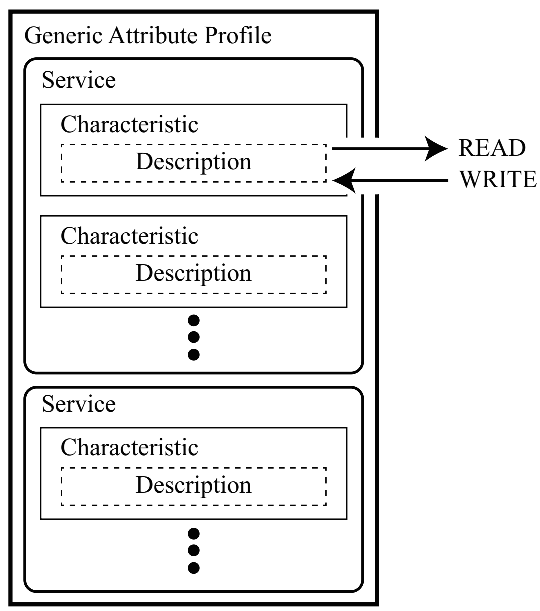

Fig. 2. GATT data hierarchy.

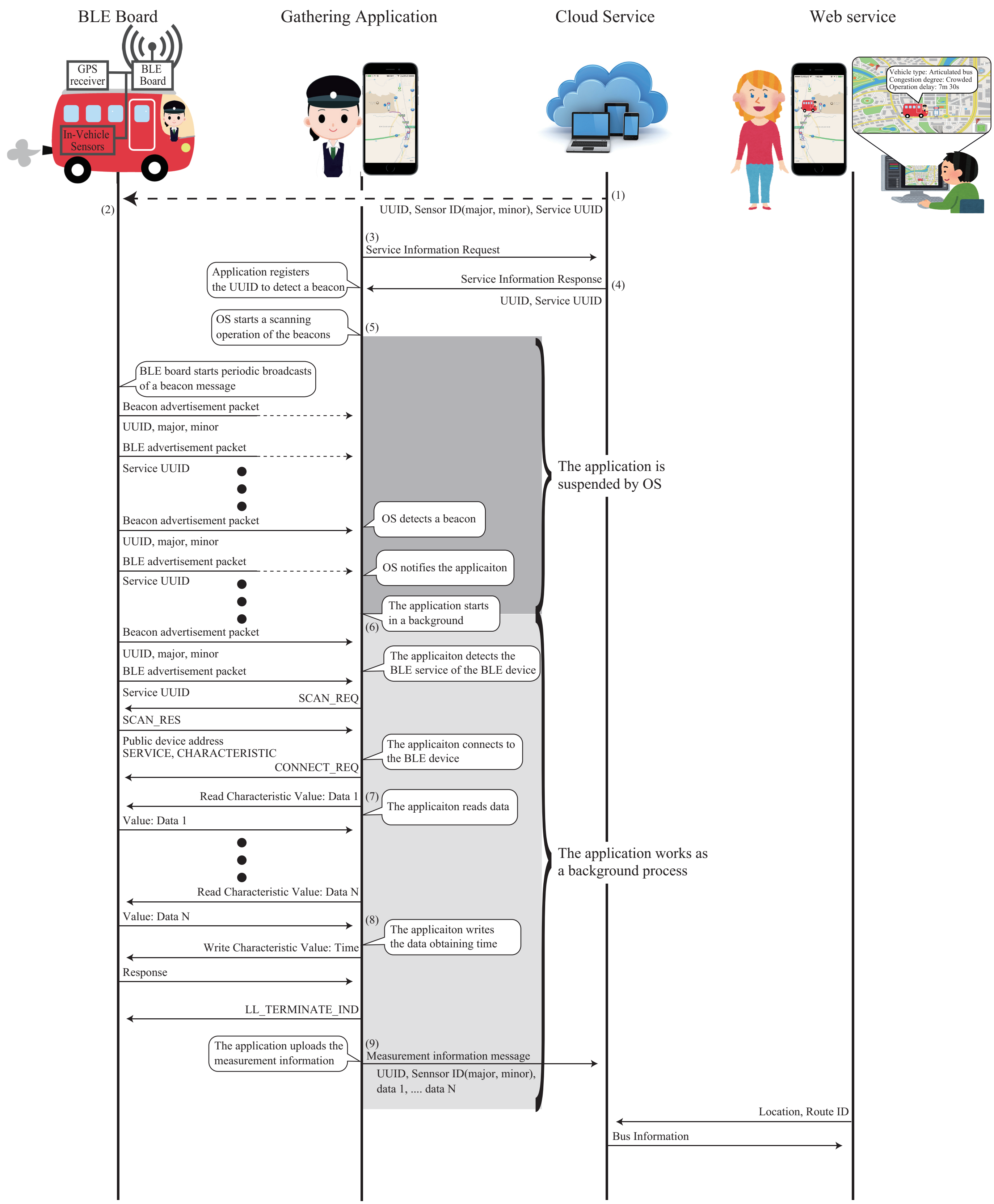

Fig. 3. Signaling process.

2.2. iBeacon

iBeacon is a protocol developed by Apple in 2013[11]. Beacons usually broadcast their identifier to portable devices. While portable devices should scan the beacons continuously, iBeacon mechanism enables iOS to scan the beacons instead of applications to reduce the consumed power. iBeacon is a BLE proximity sensing mechanism by transmitting a universally unique identifier (UUID). According to the standard, iBeacon messages include UUID, additional values called a major value and a minor value.

The distance between the transmitter of an iBeacon message and the receiver is categorized into three distinct ranges: immediate (Within a few centimeters), near (Within a couple of meters), and far (Greater than 10 meters away). The maximum range of an iBeacon transmission depends on a communication environment, and it is typically less than 100 meters.

Eddystone[12] is a BLE based beacon profile released by Google. The profile contains several frame types: Eddystone-UID, Eddystone-URL, and Eddystone-TLM. Eddystone is a more flexible standard comparing to iBeacon. Since iOS does not scan Eddystone beacons, each application should scan them in the foreground process. That means that the application may consume power to scan Eddystone beacons. Additionally, Android OS usually scan iBeacon message even if the application should scan by itself. Therefore, this paper employs iBeacon to detect a beacon device on a bus. But, the fundamental mechanism of this paper can apply to any beacon mechanisms.

3. INFORMATION GATHERING SYSTEM FOR BUS INFORMATION SHARING

3.1. System Model

Fig. 1 shows the overview of the proposed information gathering system for bus information sharing. The proposed system consists of a BLE board with the connected sensors such as a GPS receiver, vehicle information sensors, etc., a special information gathering application for a smartphone OS, a cloud service for data gathering and distribution, and a smartphone application and a web application to obtain the bus information. Since typical BLE boards support serial communication for the On Board Diagnosis second generation (OBD2) interface, detail vehicle information can also be obtained. Since the proposed system realizes automatic data collection mechanism, it utilizes beacon technologies for BLE communication [11], [12] to activate the special information gathering application. As a result, the BLE board can launch the special application on a smartphone remotely.

3.2. Signaling

Fig. 3 shows the detail signaling process of the proposed system. The BLE board uses BLE communication to advertise its service to smartphones on its bus. In our implementation, we employ iBeacon that is supported by iOS and Android OS. Therefore, the proposed system uses a UUID to identify the proposed bus location service in smartphone OSs and uses a major value and a minor value as a vehicle ID to identify each bus. The information gathering application obtains the information from the BLE board after connecting to the BLE board. It also uploads the obtained information to the cloud service.

TABLE I
IMPLEMENTATION ENVIRONMENTS.

Beacon device	Device	Nordic Semiconductor nRF51822
	BLE Version	4.1 Single-mode
	CPU	ARM Cortex-M0 32 bit
	FLASH	256KB
	SRAM	16KB
	TX power	-20dBm to +4dBm
	TX current consumption	10.5A with 0dBm
	Low power mode	Deep sleep: 1.2μA Hibernate: 4.2nA
	ADC	8 channels
	GPIO	31 Pins
	Serial	SPI, I2C, UART
Smartphone	Device	iPhone 6
	OS	iOS 11
Cloud Service	OS	Cent OS 7
	Web Server	Apache
	CGI	PHP
	DB Server	MySQL

(1) Assignment of service parameters
The proposed system uses the GATT communication and iBeacon to realize an automatic data collection mechanism. Therefore, some parameters for the GATT communication and iBeacon should be assigned to each sensor device before installing the sensor device. iBeacon uses a UUID for detecting a beacon and major and minor values for detecting a sensor device. Therefore, the cloud service assigns a specific UUID for the proposed service to trigger the specific information gathering application. Additionally, it also assigns a specific parameter with the combination of major and minor values as a sensor ID for recognizing the sensor device. This parameter information should be transferred to a manufacturing company of beacon devices beforehand.

(2) Setting of service parameter
The manufacturing company initializes each sensor device with the assigned parameters. The major and minor values are linked to sensor devices' information to recognize each sensor device.

(3) Requesting service information
The information gathering application should register a UUID to ask the iOS to scan the dedicated iBeacon message instead of the application. Additionally, it also requires a service UUID for GATT communication. It requests the information to the cloud service.

(4) Obtaining the UUID
The cloud service replies the service UUID for GATT communication and the UUID for iBeacon detection to the requested application. Therefore, the information

TABLE II
PROCESSING PERIOD.

Discovery of BLE board	765 ms
Scan request	614 ms
Connection to BLE board	455 ms
Read characteristic	64 ms for each
Uploading	19 ms

gathering application can obtain the parameters. Then, it registers the obtained UUID to the iOS to scan a dedicated iBeacon message. The iOS starts to scan the dedicated iBeacon message including the UUID. Since the scanning process of the dedicated iBeacon message is optimized by the iOS, the scanning application can perform the dedicated function with low power consumption. Finally, the information gathering application changes the status from a background processing to a suspended status to reduce the consumed power. After that, iOS can scan a beacon related to the registered UUID continuously.

(5) Starting as a background processing
Each beacon device transmits iBeacon messages periodically. Each iBeacon message includes the UUID and the vehicle's ID with the major and minor values. When the iOS detects the iBeacon message from the beacon device, it checks the received UUID to the registered UUID list. When it finds the registered UUID, it triggers the dedicated application related to the received UUID. After that, the triggered application can start as the background processing. The processing period is limited for up to 10 seconds due to the limitation of iOS. The proposed system assumes that the application should finish downloading the measured information from the beacon device and uploading it to the cloud service.

(6) Scanning of the BLE device
Since the beacon device transmits a BLE advertisement packet after transmitting of the iBeacon message, the application should start scanning of the BLE advertisement packet. When the application receives the BLE advertisement packet from the beacon device, it transmits the SCAN_REQ packet. The beacon device replies the SCAN_RES packet as the response to the SCAN_REQ packet. Since the SCAN_REQ packet coveys the service UUID of the beacon device and the parameters for GATT communication, the application tries to connect to the beacon device by transmitting CONNECT_REQ packet.

(7) Obtaining the sensor data
The beacon device stores measurement information as some characteristic values of GATT. Therefore, the application receives the measurement information by reading dedicated characteristic values. Since GATT provides a flexible definition of data structures, the system should define each characteristic according to measurement information.

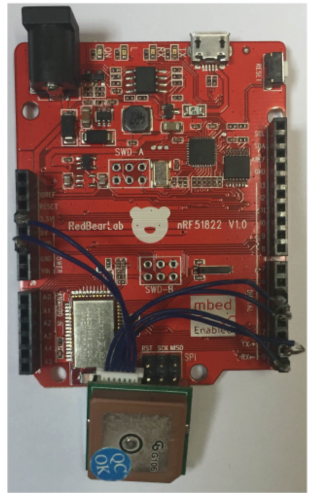

Fig. 4. BLE board.

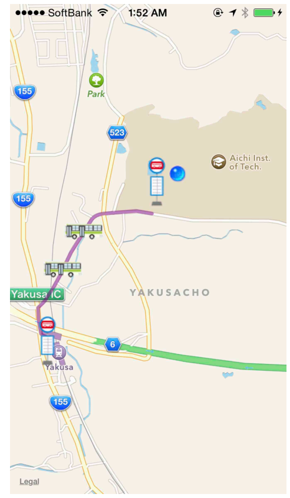

Fig. 5. Smartphone application.

(8) Storing the obtaining time
When the application obtains the whole measurement information from the beacon device, it stores the obtaining time by writing to the dedicated characteristic value.

(9) Uploading the information
After the application obtains the measurement information, it also uploads the information to the cloud service. The operation of these processes is performed in background processing. Therefore, smartphone users do not care about the application. Finally, the application also changes the status from a background processing to a suspended status to reduce the consumed power.

3.3. Implementation

Tab. I is the implementation environments. We have developed a BLE board with an ARM-based microcomputer board. We employed Nordic Semiconductor nRF51822 System on Chip (SoC) and a GPS receiver connecting through UART. Since RedBearLab nRF51822 supports mbed environment and Arduino environment, we have employed a mbed environment to develop the BLE board. The board is built around a 32-bit CPU with 256kB flash + 16kB RAM. The BLE board should store its configuration parameters(UUID, Major, Minor) for iBeacon function and parameters for GATT communication. Therefore, these parameters are installed when the developed program is written into the microcontroller. The BLE board transmits an iBeacon message periodically to trigger the information gathering application. Additionally, it also transmits a BLE advertisement packet after transmitting of the iBeacon message. Since it takes time to restart the information gathering application by an iBeacon message, we have configured the duration between the transmissions of an iBeacon message and a BLE advertisement packet.

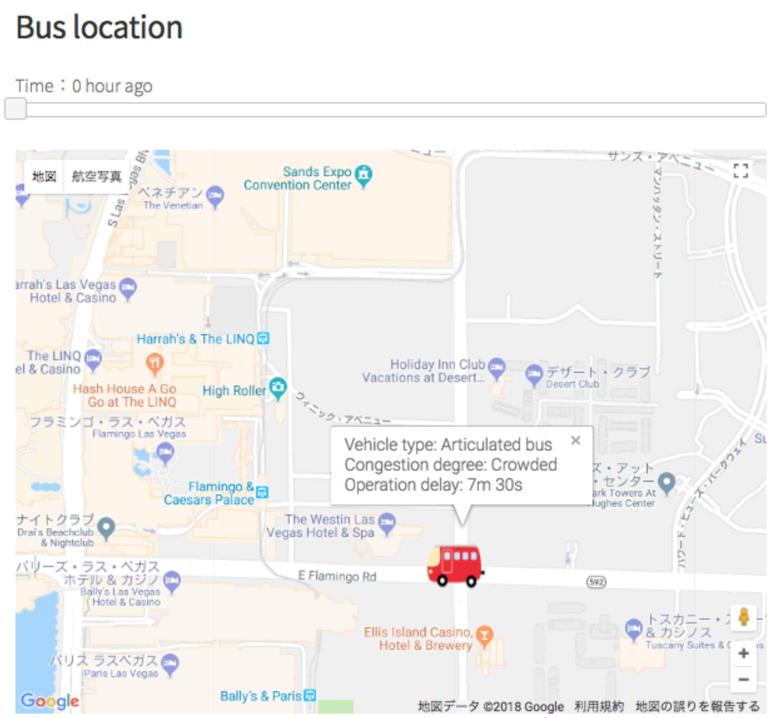

Fig. 6. Web application view.

As the example sensor data, we have connected the GPS module to the BLE board. Therefore, the BLE board receives the location data from the GPS module through UART communication. The received information is stored into each characteristic value.

We have developed the prototype information gathering application for iOS 11. As the iBeacon detection function, we have employed CLBeaconRegion. Tab. I shows the numerical result of the processing period. We have evaluated the prototype application 10 times, and take an average of each processing period. From the measurement results, we have found that the smartphone application can upload the bus location through the BLE board. Additionally, we have found that the smartphone application can perform whole functions: measurement and uploading of bus location within 10 [s] that is the allowable period of background processing in iOS. Fig. 5 shows the demonstration view of the prototype application. The application can access the cloud service to obtain bus information in the dedicated area. Therefore, the prototype application can work as both the information gathering function and the bus location function.

The cloud service has been developed as a PHP-based web application. Therefore, almost all web service can implement our cloud application. The functions of the cloud service are the management of routes, bus information, bus location, and extra information. Fig. 6 is the demonstration view of the developed web application.

4. Conclusion

This paper has proposed a collaborative mechanism with a BLE board and smartphones to realize an information gathering system for bus information sharing. The proposed system can realize flexible data collecting for various information by a microcomputer board and reasonable cost for installation and operation by smartphones. The proposed mechanism employs BLE devices that trigger our special application for smartphones. Therefore, the special application can be automatically launched when a beacon message arrives on the smartphone. Therefore, smartphone users do not care about the data collection process. In the experimental evaluation, we have developed the proposed system on a typical wireless SoC module with a GPS receiver and the iOS system. The experimental results demonstrated that the developed system works well as an information gathering system.

Acknowledgment

This work is supported in part by Grant-in-Aid for Scientific Research (B)(15H02697) and (C)(17K00142), Japan Society for the Promotion of Science (JSPS), and the Cooperative Research Project Program of the Research Institute of Electrical Communication, Tohoku University.

REFERENCES

[1] P. Mohan, V. N. Padmanabhan, R. Ramjee, "Nericell: rich monitoring of road and traffic conditions using mobile smartphone devices," The 6th ACM conference on Embedded network sensor systems (SenSys '08), November 2008.

[2] S. Foell, G. Kortuem, R. Rawassizadeh, M. Handte, U. Iqbal, and P. Marrón, "Micro-navigation for urban bus passengers: using the internet of things to improve the public transport experience," The First International Conference on IoT in Urban Space, pp. 1–6, October 2014.

[3] N. Ronald, R. Thompson, J. Haasz, and S. Winter, "Determining the Viability of a Demand-Responsive Transport System under Varying Demand Scenarios," The Sixth ACM SIGSPATIAL International Workshop on Computational Transportation Science, November 2013.

[4] B. Ferris, K. Watkins, and A. Borning, "OneBusAway: results from providing real-time arrival information for public transit," CHI '10 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 1807–1816, April 2010.

[5] J. Zimmerman, A. Tomasic, C. G. D. Yoo, C. Hiruncharoenvate, N. R. T. Y Huang, and A. Steinfeld, "Field trial of Tiramisu: crowd-sourcing bus arrival times to spur co-design," The SIGCHI Conference on Human Factors in Computing Systems, pp. 1677–1686, May 2011.

[6] N. D. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, A. T. Campbell, "A survey of mobile phone sensing," IEEE Communications Magazine, Vol. 48 No. 9, September 2010.

[7] J. Biagioni, T. Gerlich, T. Merrifield, and J. Eriksson, "EasyTracker: automatic transit tracking, mapping, and arrival time prediction using smartphones," The 9th ACM Conference on Embedded Networked Sensor Systems, pp. 68–81, November 2011.

[8] A. Thiagarajan, J. Biagioni, T. Gerlich, and J. Eriksson, "Cooperative transit tracking using smart-phones," ACM SenSys, pp. 85–98, 2010.

[9] P. Zhou, Y. Zheng, and M. Li, "How Long to Wait? Predicting Bus Arrival Time With Mobile Phone Based Participatory Sensing," IEEE Transactions on Mobile Computing, Vo.13, No. 6, pp. 1228–1241, October 2013.

[10] K, Farkas, A. Z. Nagy, T. Tomas, R. Szabo, "Participatory sensing based real-time public transport information service," 2014 IEEE International Conference on Pervasive Computing and Communications Workshops, March 2014.

[11] https://developer.apple.com/ibeacon/

[12] https://developers.google.com/beacons/