

# A Petri Net Model of Location-Based Content Providing Systems

Gyuhyeon Jo<sup>1</sup> and Jaegeol Yim<sup>1\*</sup>

<sup>1</sup>*Dept. of Computer Engineering, Dongguk University at Gyeongju,  
Gyeongbuk, 38066, Korea  
yim@dongguk.ac.kr*

## **Abstract**

*In software system development, it is well known that investing efforts of proving correctness of the design of the system before implementing the software system is much more cost effective than skipping this stage and directly jump into the implementation stage. The process of proving correctness of the design usually starts with formalization of the design. Among the countless formal representation methods, this paper uses Petri net that is a mathematical and graphical modeling tool. Location based service (LBS) has been studied for a long time and most of people enjoy LBS everybody. This paper presents a Petri net model of a location-based content providing system, a type of LBS. Then, the minimum cycle time of the developed Petri net will be analyzed. The minimum cycle time of the Petri net is close to the response time of the LBS system that is one of the important factors specified in system requirements.*

**Keywords:** *Petri net, Location-Based Service, Minimum Cycle Time, Hierarchical Petri Net, Verification*

## **1. Introduction**

Software verification tries to ensure that your final software matches the original design, *i.e.* you built your software according to the prescribed specifications. Such specifications should provide a complete description of the behavior of the system to be developed, including a library of use cases that describe all possible interactions between end users and the software [1]. The process of software verification starts with formal description of the prescribed specifications. Petri net has been widely used to formally describe the prescribed specification [2-3].

A location based service (LBS) provides useful information to the users based on the geographic location the user designates or the user is currently located. LBS is so useful to the people and most of the people use LBS every day. Vehicle navigation system is an example LBS that every driver uses every day [4].

A location-based content providing system automatically recognizes the current location of the user and recommends the videos that are closely related to the spot where the user is located. This paper introduces a Petri net model of a location-based content providing system and analyzes the minimum cycle time of the model.

## **2. Related Researches**

The smart cross platform's objective is to effectively execute game and multimedia contents as a contents execution platform developed by the research team on various OS and devices. Also it does not depend on specific development language and supports numerous programming languages.

---

\* Corresponding Author

The authors of [5] introduced an adaptive extended Kalman filter (AEKF) algorithm that locates a moving target using both the time difference of arrival and the frequency difference of arrival. The Kalman filter process repeats the time update phase and the measurement update phase. The noise covariance is not changed during the process. Contrarily, the noise covariance of AEKF is updated at every time update and measurement update.

A GPS-based ticket reservation system was introduced in [6]. The client part of this system runs on a smartphone. The smartphone app receives GPS signals and sends them to the location information server that is connected to the reservation database server. A user can make a request for reservation of a service with this app. The location information server determines whether the request for reservation should be granted or not considering the number of people on the waiting queue for the requested service. In addition to the reservation service, it also provides the real-time partner checking service (let the people know the current locations of their partners, young children for example), the real-time emergency call service, the location-based real-time event notice service, and so on.

Current recommender systems can be classified into collaborative filtering recommendation, content-based recommendation, and knowledge-based recommendation. In a collaborative filtering recommendation system, a critic rating matrix is maintained in the server. Referring to the critic rating matrix, we can find who likes which content. This system finds a set of persons,  $A$ , who are most similar to the user and recommends the content that the persons in  $A$  like to the user. A content-based recommendation system recommends items to the users considering their past experience. The authors of [7] proposed a location-based recommendation system that delivers information that is related to the current location of the user. This system determines the current location of the user with the assisted global positioning system (A-GPS). Referring to the current location of the user, the user profile database, the tourism database, and usage log, the system extracts items to be recommended. The strategy employed by this recommendation system is a hybrid of the content based recommendation and the collaborative filtering recommendation.

The WiFi fingerprint-based indoor localization calculates the Euclidean signal distance between the signals collected at the test location and the signals collected at the reference point. The authors of [8] proposed an alternative way of calculating the signal distance and improved the accuracy of the fingerprint-based indoor localization.

The readers of this paper must be familiar with the basic concepts regarding Petri nets that appear in [9]. Petri nets, a mathematical and graphical modeling tool, have been widely used in various fields since the concept was first introduced in Petri's 1962 dissertation. A Petri net consists of places, transitions, arcs, a weight function, and an initial marking, as outlined in Table 1. A transition in a Petri net model usually represents an event or operation, a place is associated with a resource to be consumed by an operation or produced as the result of an operation, and an arc represents the flow of resources. The weight function associates an arc with the number of resources to be consumed or produced. The initial marking represents the amount of resources initially available. One resource unit is represented by a black dot, or a token, and a marking in a Petri net denotes the number of tokens located to each of the places.

**Table 1. A Definition of a Petri Net**

|   |
|---|
| <p>[Def. 1] A Petri net is a 5-tuple, <math>PN = (P, T, F, W, M_0)</math> where:<br/> <math>P = \{p_1, p_2, \dots, p_m\}</math> is a finite set of places,<br/> <math>T = \{t_1, t_2, \dots, t_n\}</math> is a finite set of transitions,<br/> <math>F \subseteq (P \times T) \cup (T \times P)</math> is a set of arcs (flow relation),<br/> <math>W: F \rightarrow \{1, 2, 3, \dots\}</math> is a weight function,<br/> <math>M_0: P \rightarrow \{0, 1, 2, \dots\}</math> is the initial marking,<br/> <math>P \cap T = \Phi</math> and <math>P \cup T \neq \Phi</math>.</p> |
|---|

The behavior of a system can be described in terms of system states and their changes. In order to simulate the dynamic behavior of a system, a state or marking  $M$  in a Petri net is changed to  $M'$  according to the following transition firing rule:

1. A transition  $t$  is said to be enabled if each input place  $P$  of  $t$  is marked with at least  $W(p,t)$  tokens.

2. An enabled transition may fire.

A firing enabled transition  $t$  removes  $W(p,t)$  tokens from each input  $P$  of  $t$ , and adds  $W(t,p)$  tokens at each output  $P$  of  $t$ .

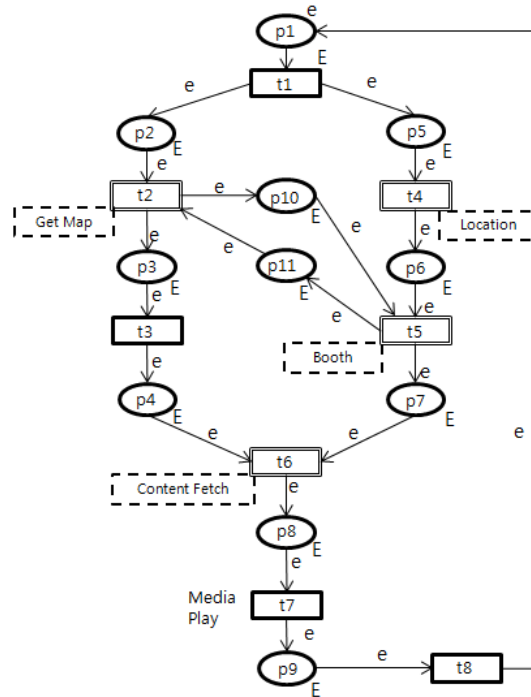
Petri nets grow quickly for modeling complex systems, and the built models are difficult to be managed and understood. In order to model complex systems, colored Petri nets should be used because colored Petri nets provide parameterized and compact representations of complex systems by folding similar components into one and differentiating them using different colors. The authors of [10] presented a colored stochastic Petri net model of a complex biological system.

The authors of [11] discussed a unified modeling language (UML) description of the leave home scenario and converted the UML description into a colored Petri net. Then, it verified the scenario with the coverability graph of the Petri net.

### 3. Petri Net Model of a Location Based System

This paper develops a Petri net model of a location based content providing system. The objective of developing the Petri net model is to prove that the response time of the system is quicker than the response time required by the user. The system response time is close to the minimum cycle time of the Petri net. Therefore, the minimum cycle time of the developed Petri net will be analyzed.

Our Petri net model of the location based content providing system is shown in Figure 1. We used CPNTools to build the Petri net. As determined by the current location of a user, a location based content providing system selects the appropriate content and displays the content on the user's mobile device, *i.e.*, a smart phone.



**Figure 1. The Hierarchical Petri Net Model of a Location Based Contents Providing System**

In order to achieve this end, the system starts the “Get Map” and “Location” web services concurrently. The “Get Map” web service reads the electronic map file (for example, an AutoCAD file) requested by the caller and returns the essential part of the file with which an image of the map can be drawn. The “Location” web service determines the location of the mobile device when the sensor data captured by the mobile device is given and returns the location information [13].

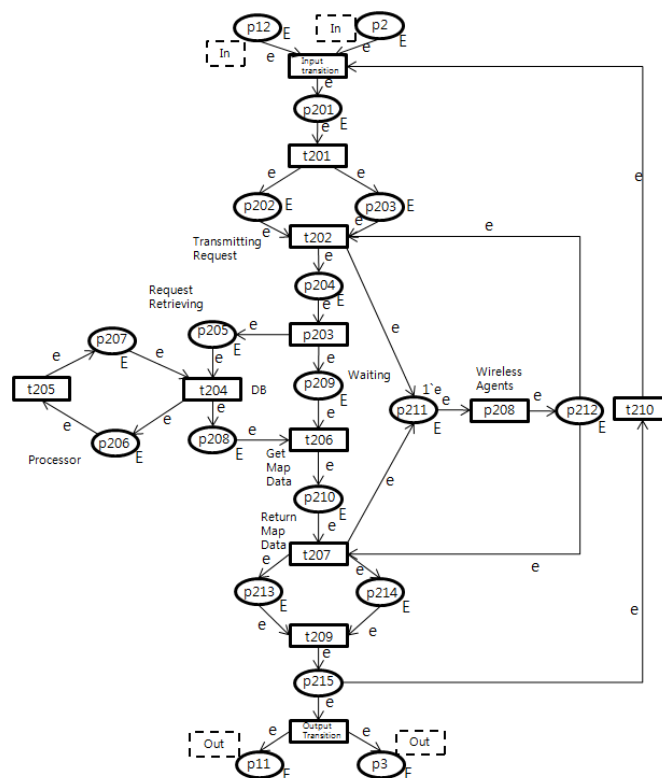
Once the map data and the location data are given, it invokes the local function “Booth”, which determines if there is any point of interest around the mobile device, or if it is located inside a booth. If the case is true, then the mobile device invokes the “Content Fetch” web service to obtain the appropriate content and displays it. This whole process repeats endlessly while this system runs.

In timed nets, a transition is associated with a delay time. If delay time  $d_i$  is associated with transition  $t_i$ , when  $t_i$  is enabled,  $a_{ji}$  tokens will be reserved in place  $P_j$  for at least  $d_i$  sec before their removal by firing  $t_i$ , where  $a_{ji}$  is the weight of the arc from  $P_j$  to  $t_i$  [12].

A cycle time is defined as the time needed to complete a firing sequence from the initial marking back to the same marking after firing each transition at least once. This implies that if a Petri net is not consistent then it does not possess a cycle time. We denote the cycle time as  $\tau$ . It is well known that a lower bound of  $\tau$ , *i.e.* the minimum cycle time, is given by:

$$\tau_{\min} = \text{Max}_k \left\{ y_k^T (A^-)^T D x / y_k^T M_0 \right\} \quad (1)$$

where,  $y_k$  is an S-invariant,  $A^- = [a_{ij}^-]_{n \times m}$ ,  $x$  is a positive T-invariant,  $a_{ij}^- = w(p_j, t_i)$ ,  $D$  is the diagonal matrix of  $d_i$ ,  $i=1, 2, \dots, n$ , and  $M_0$  is the initial marking [12].



**Figure 2. The Refinement Operator Results Applied to  $t_2$  (Get Map)**

Given a hierarchical Petri net, as shown in Figure 1, our minimum cycle time analysis process proceeds as follows:

Vector  $x^T = (11111111)$  is the minimum positive T-invariant of the Petri net shown in Figure 1. It has the following three minimal support S-invariants:

$$y_1^T = (00000000011)$$

$$y_2^T = (11110001100)$$

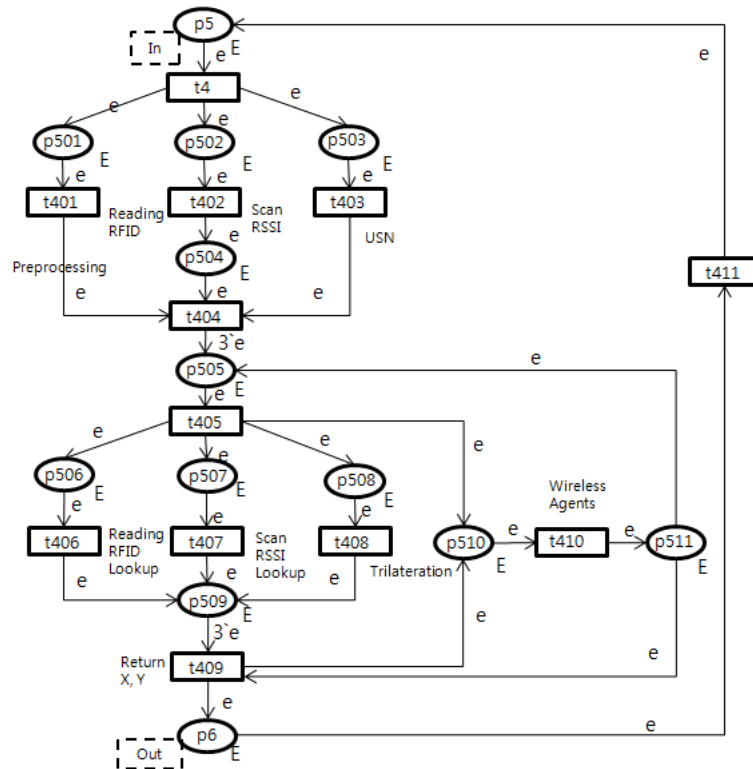
$$y_3^T = (10001111100)$$

Applying (Equation 1), we obtain the following minimum cycle time:

$$MAX \{d_2 + d_5, d_1 + d_2 + d_3 + d_6 + d_7 + d_8, d_1 + d_4 + d_5 + d_6 + d_7 + d_8\} \quad (2),$$

where  $d_i$  is the delay time of  $t_i$ . CPNTools does not support subscripts so the subscripts are written in ordinary fonts in the figure.

Figure 2 shows the sub Petri net associated to  $t_2$  (Get Map). If a mobile device requests map information by sending the name of the area, the “Get Map” web service retrieves the file of the requested map from the database and extracts the geometrical information, such as a line, polyline, circle, *etc.*, from the file and returns the geometrical information to the mobile device. In Figure 2,  $t_{205}$  represents the retrieval operation and  $p_{206}$  represents the DB server processors handling the map database. Transitions  $t_{202}$  and  $t_{207}$  represent the wireless communication between the DB server and the mobile device and places  $p_{211}$  and  $p_{212}$  represent the wireless agents. If there are many wireless agents then this web service can communicate with many mobile devices at the same time. Similarly, if there are many processors in the DB server, then it can serve many retrieval requests at the same time. In this figure  $t_{210}$  is the feedback transition added to make the consistent refinement.



**Figure 3. The Application of the Refinement Operator to  $t_4$  (Location)**

Vector  $x^T = (111111121)$  is the minimum positive T-invariant of the Petri net shown in Figure 2. It has following three minimal support S-invariants:

$$y_1^T = (00000000011000)$$

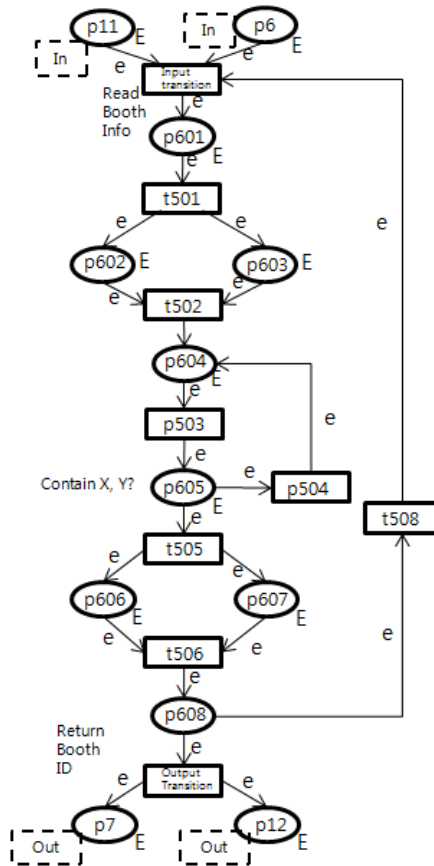
$$y_2^T = (000001100000000)$$

$$y_3^T = (211210011000112)$$

Applying (Equation 1), we obtain the following minimum cycle time:

$$MAX \left\{ \begin{array}{l} (d_{202} + d_{207} + 2d_{208})/k, \\ (d_{204} + d_{205})/r, \\ (2d_{201} + 2d_{202} + 2d_{203} + d_{204} + 2d_{206} + 2d_{207} + 2d_{209})/2 \end{array} \right\} \quad (3)$$

The delay time of the feedback transition  $t_{210}$  should be 0 so it can be safely omitted from the cycle time. In (Equation 3),  $(d_{202} + d_{207} + 2d_{208})/k$  represents the time needed for communication where  $k$  is the number of wireless communication agents.  $(d_{204} + d_{205})/r$  represents the time needed for retrieving the map file from the database where the DB server is equipped with  $r$  processors.  $(2d_{201} + 2d_{202} + 2d_{203} + d_{204} + 2d_{206} + 2d_{207} + 2d_{209})/2$  is the time of the interval between the point of requesting the web service and the point of receiving the results of the service.



**Figure 4. The Application of the Refinement Operator to  $t_5$  (Booth)**

The Petri net shown in Figure 6 is the sub-Petri net associated to  $t_4$  (Location). If a mobile device collects and sends RFID, RSSI and/or USN data to an indoor positioning web service then the web service returns the estimated location of the mobile device. Considering the description of Figure 2, the meaning of this model should be clear so a detailed description is omitted.

Vector  $x^T = (11111111121)$  is the minimum positive T-invariant of the Petri net shown in Figure 3. It has only two minimal support S-invariants as follows:

$$y_1^T = (000000000110)$$

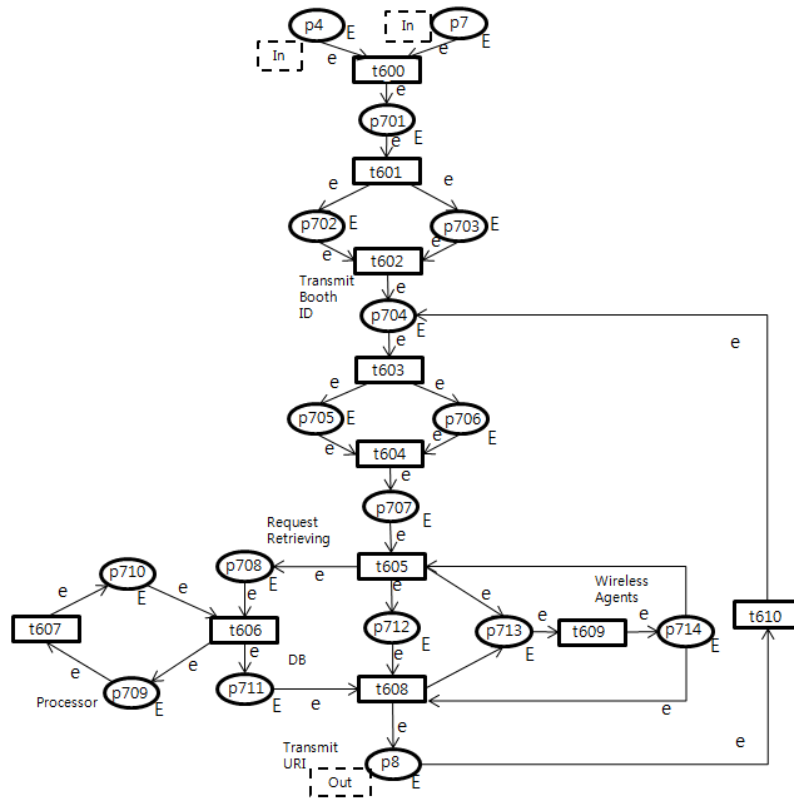
$$y_2^T = (311111111003)$$

Applying (Equation 1), we obtain the following minimum cycle time:

$$MAX \left\{ \begin{array}{l} (d_{405} + d_{409} + 2d_{410})/k, \\ (3d_{400} + d_{401} + d_{402} + d_{403} + d_{404} + 3d_{405} + d_{406} + d_{407} + d_{408} + 3d_{409} + 3d_{411})/3 \end{array} \right\} \quad (4)$$

A detailed description of this equation is similar to the description of (Equation 3) and so is omitted.

The Petri net shown in Figure 4 is the sub-Petri net associated to  $t_5$  (Booth). This Petri net describes the process of determining the points of interest around the mobile terminal. The process takes the map (from the “Get Map” web service) and the location (from the “Location” web service) as its input and for each of the areas where a point of interest is positioned checks to see if the location lies within the area. If it finds any area where the location is included then it returns the point of interest associated with the area, otherwise it returns NIL. We omitted the feedback transition in this exercise.



**Figure 5. The Application of the Refinement Operator on  $t_6$  (Content Fetch)**

Vector  $x^T = (1121111)$  is the minimum positive T-invariant of the Petri net shown in Figure 4. It has four minimal support S-invariants as follows:

$$y_1^T = (1101110)$$

$$y_2^T = (1011110)$$

$$y_3^T = (1101101)$$

$$y_4^T = (1011101)$$

In this case, we get the same cycle time for all of the four S-invariants:

$$\text{MAX} \{d_{501} + d_{502} + 2d_{503} + d_{504} + d_{505} + d_{506}\} \quad (5)$$

The Petri net shown in Figure 5 is the sub-Petri net associated to  $t_6$  (Content Fetch). The mobile device invokes the web service of “Contents Fetch” web service providing the ID of the point of interest obtained from “Booth.” The web service then retrieves the content associated with the ID and returns the content to the mobile device via the wireless communication network. In this Petri net, we omitted the out transition because it has only one out socket.

Vector  $x^T = (1111112)$  is the minimum positive T-invariant of the Petri net shown in Figure 5. It has six minimal support S-invariants with four of them having the same cycle time. The minimum cycle time of the net is:

$$\text{MAX} \left\{ \begin{array}{l} (d_{602} + d_{606} + 2d_{607})/k, \\ (d_{604} + d_{605})/r, \\ d_{601} + d_{602} + d_{603} + d_{604} + d_{606} + d_{608} \end{array} \right\} \quad (6)$$



We can determine the minimum cycle time of the abstract Petri net shown in Figure 1:

$$MAX \left\{ \begin{array}{l} d_2 + d_5, \\ d_1 + d_2 + d_3 + d_6 + d_7 + d_8, \\ d_1 + d_4 + d_5 + d_6 + d_7 + d_8 \end{array} \right\} = MAX \left\{ \begin{array}{l} (Eq3) + (Eq5), \\ d_1 + (Eq3) + d_3 + (Eq6) + d_7 + d_8, \\ d_1 + (Eq4) + (Eq5) + (Eq6) + d_7 + d_8 \end{array} \right\} \quad (7)$$

The left hand side of the expression equates to (Equation 2). We derived the right hand side of the expression from the left hand side by replacing the delay times with the minimum cycle times of the consistent refinements. For example, the delay time of the abstract transition  $t_2$  denoted by  $d_2$  is replaced by (Equation 3), which is the minimum cycle time of the consistent refinement of  $t_2$ .

## 4. Conclusions

A Petri net model of the location-based content providing system is introduced in this paper. The Petri net is a hierarchical and timed. In hierarchical Petri net, transitions can be abstract. An abstract transition represents a complex process and can be replaced by a sub-Petri net. Transitions of a timed Petri net is associated with delay time. This paper described the process of calculating the minimum cycle time of a hierarchical timed Petri net.

## Acknowledgments

This research was supported by the Dongguk University Research Fund of 2016, by the Ministry of Education (NRF-2011-0006942), and by the Ministry of Knowledge Economy (10037393).

## References

- [1] A. Iglesias, "Pure Petri Nets for Software Verification and Validation of Semantic Web Services in Graphical Worlds," *IJFGCN*, vol. 3, no. 1 (2010), pp. 33-46.
- [2] J. Yim, J. Joo and Y. Lee, "Timed Net with Choice Probability and Its Minimum Cycle Time-The Case of Location Based Service," *Expert Systems with Applications*, vol. 31, no. 2, (2006), pp. 221-230
- [3] J. Yim: "A Minimum Cycle Time Analysis for Hierarchical Petri Nets," to be published
- [4] G. Lee and J. Yim, "A Review of the Techniques for Indoor Location based Service," *IJGDC*, vol. 5, no. 1, (2012), pp. 1-22.
- [5] D. Kim, J. Ha and K. You, "Adaptive Extended Kalman Filter Based Geolocation Using TDOA/FDOA," *IJCA*, vol. 4, no. 2 (2011), pp. 49-58.
- [6] S. Kang, H. Kang, J. Kim, H. Lee and J. Lee, "A Study on the Mobile Communication Network with Smart Phone for Building of Location Based Real Time Reservation System," *IJMUE*, vol. 7, no. 2 (2012), pp. 17-36.
- [7] W. Husain and L. Dih, "A Framework of a Personalized Location-based Traveler Recommendation System in Mobile Application", *IJMUE*, vol. 7, no. 3 (2012), pp. 11-18.
- [8] J. So, J. Lee, C. Yoon and H. Park: "An Improved Location Estimation Method for Wifi Fingerprint-based Indoor Localization," *IJSEIA*, vol. 7, no. 3, (2013), pp. 77-86.
- [9] T. Murata: "Petri Nets: Properties, Analysis and Applications," *Proceedings of the IEEE*, vol. 7, no. 4 (1989), pp. 541-580.
- [10] F. Liu and M. Yang, "Colored Stochastic Petri Nets for Modeling Complex Biological Systems," *IJHIT*, vol. 6, no. 5 (2013), pp. 11-24.
- [11] X. Niu and Z. Wang, "A Smart Home Context-aware Model Based On UML and Colored Petri Net," *IJSH*, vol. 10, no. 1 (2016), pp. 101-114.
- [12] J. Yim, "A Review of Petri Net Theories," *Advanced Science and Technology Letters Vol.133 (Information Technology and Computer Science 2016)* (2016), pp. 95-98.

- [13] G. Jo, J. Yim, "An Example Petri Net Model of a Location-Based Service", Asia-pacific Proceedings of Applied Science and Engineering for Better Human Life, vol. 9, (2016), pp. 70-74.