A Middleware for Cyber Physical Systems in an Internet of Things Environment: Case of for Mobile Asset Tracking

Muthoni Masinde(✉) and Admire Mhlaba

Unit for Research on Informatics for Droughts in Africa(URIDA),
Central University of Technology, Free State, Private Bag X20539,
Bloemfontein 9300, South Africa
muthonimasinde@gmail.com, yaddly@gmail.com

Abstract. The upsurge in Cyber Physical Systems (CPSs) has made researchers conclude that these systems have the potential of rivalling the contribution of the Internet. Driving this wave is the emergence of miniaturized, cheaper and readily available location-based hardware devices. One of the main applications of CPSs is mobile asset tracking system whose roles are to monitor movements of a mobile asset and to track the object's current position. Localization accuracy of these systems is one of the key performance indicators. This is usually maximised through the introduction of extra hardware devices. The drawbacks with this approach include restriction of the system's application only to one domain, introduction of extra cost to the overall system and introduction of a single point of failure. Conversely, the Internet of Things (IoT) paradigm facilitates coalescing of diverse technologies through which locus data can be extracted in cost-effective and robust way. The challenge is the lack of a dependable and responsive middleware that is capable of handling and servicing such devices. We present a solution to this problem; a middleware designed around In-lining approach that acts as an insulator for hiding the internal workings of the system by providing homogenous and abstract environment to the higher layers. The evaluation of laptop tracking and monitoring system prototype was carried out through implementation of a middleware that integrates diverse IoT components in a university environment.

Keywords: Cyber Physical Systems (CPSs)
Mobile Asset Management (MAM) · Internet of Things (IoT) · Middleware
Laptop monitoring and tracking system (LMTS)

1 Introduction

1.1 Background Information

The concept behind Cyber Physical Systems (CPSs) is the incorporation of information and data communication technologies with the physical and real worlds, especially engineering operations and tasks [1]. Although invented back in 2006 [2], there is noticeable increase in CPSs applications which is driven by the emergence of miniaturized, cheaper and readily available location-based hardware devices. This draws

R. Zitouni and M. Agueh (Eds.): AFRICATEK 2018, LNICST 260, pp. 18–31, 2019.
https://doi.org/10.1007/978-3-030-05198-3_2

parallels with other related technological advancements such as Internet of Things (IoT), Machine-to-Machine (M2 M) communication and Wireless sensor networks (WSNs). Figure 1 below shows the evolution of CPSs and their correlation with these technologies [3]. There are possibilities that CPSs may even surpass the massive contributions made by internet [2].

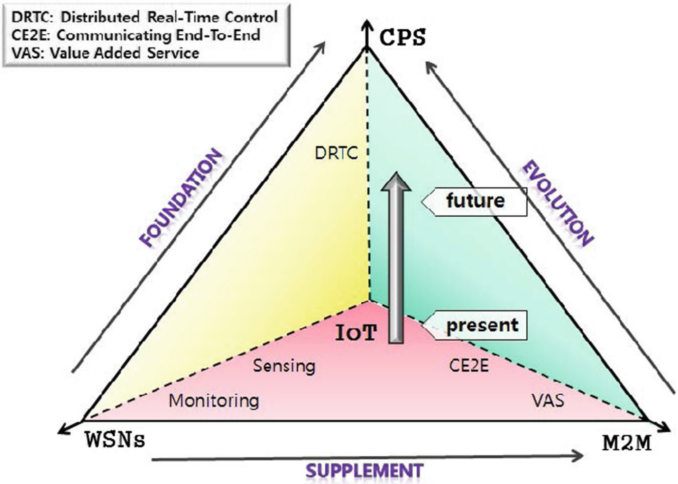


Fig. 1. IoT application range of CPSs [3]

Application areas of CPSs include smart transportation, smart cities, precision agriculture and entertainment [4]. Over the last decade, CPSs frameworks have been developed to address the following aspects: complexity, adaptability, safety, reliability and maintainability. In [4, 5] for instance, the focus is on adaptability. One area where CPSs have attained some maturity is in mobile asset tracking systems or mobile asset management systems (MAMs). Their application domains include patient monitoring, equipment management and emergency management [6]. MAMs implantation involve attaching a node to a mobile object to monitor the object's movement and current location real-time [7]. This is achieved through location-based services.

The requirements for location-based systems introduce opposing performance metrics such as localization accuracy, precision, complicity, cost effectiveness, low power consumption and tiny size, portability across domains, robustness and scalability (variable number of nodes) [8]. In particular, accuracy factor is inversely proportional to the number of nodes that can be supported simultaneously [9]. Despite this, when it comes to tracking our valuable assets, localization accuracy ranks high because it increases chances of recovering the asset.

Most implementers of asset tracking systems tend to achieve this accuracy by introducing extra hardware [10]. Global Positioning System (GPS), General Packet Radio Service (GPRS) and Radio Frequency Identifier (RFID) technology provide such devices; however, the drawback with this approach is that the cost of the solution is considerably increased and the application domain is highly restricted. Besides, such applications introduce a single point of failure or bottleneck; for instance, GPS devices tend to fail when indoors and their high demand for power makes them inappropriate for use for small mobile assets and battery powered devices.

On the contrary, CPSs and Internet of Things (IoT) enables the unification of multiple technologies (the Internet, mobile phones, RFID readers/tags, Bluetooth, Wi-Fi, ZigBee, GPRS/GPS etc.) through which location information of heterogeneous objects can be obtained in cost-effective and robust way. Such asset tracking systems are usually composed of an array of various objects interlinked by diverse communication technologies. Each of these devices function through local and/or remote communication with the real world or other devices and systems. However, the difficult of maintaining a reliable and reactive middleware that is capable of handling and servicing such devices, process volumes of data without compromising responsiveness is still eminent. The very nature of CPSs and IoT introduces a number of challenges: first, the number of nodes involved can quickly grow into (tens of) thousands, hence increasing contention for limited resources (especially the bandwidth). Secondly, the introduction of mobile nodes immediately introduces the need for location-awareness in the communication, which is still difficult in the existing communication protocols [9].

An asset is anything that has intrinsic or substantial value to a business or individual entity. Assets well managed by asset management systems, can among other things lead to sound financial gains and mitigate risks. The development of ISO standard (ISO 55001:2014) is indicative of the importance of asset management systems along with their regulated implementation [11]. Current mobile asset tracking systems are expensive and inefficient – this is especially due to the cost transferring huge amounts of data that is required in tracking assets' position and velocity [12].

The intensified use of mobile devices such as smartphones and tablets has workers around the globe increasingly becoming mobile - they to do their work at the office, at home, and while travelling. This has resulted to the anytime, anywhere information workers - those who use three or more devices, working from multiple locations, and use many applications [13]. Consequently, the traditional asset management and tracking systems have to be re-designed to cater for this as well as for the "bring your own device" (BYOD) concept. In the meantime, the availability of these devices has led to an increase in their (devices) loss through theft. This increase in larceny is somewhat motivated by the fact that laptops (including iPads and tablets) are miniature and easy to conceal and pocket away. In addition, filched devices carry a remarkable resale value on the informal market and are conveniently disposed of online, using platforms such as Gumtree, cheaply and anonymously. The difficulties in tracking and tracing the physical location of stolen mobile devices can be attributed as the primary reason for the surge in theft. In an attempt to annihilate this growing calamity, many solutions have been developed, however several small and medium-sized organizations are compelled to do with one due high cost of ownership. Regardless of improvements in electronic engineering and availability of miniature GPS and GPRS hardware, there remains a gap that portable computing device manufacturers need to fill with regards to the integration of tracking technologies to combat this menace.

1.2 Research Objectives

The main objective of this research was to develop a generic IoT architecture that innovatively and intelligently integrates wireless sensors, RFID tags (and readers), fingerprint readers, and mobile phones. The operation of this middleware was then evaluated using an asset monitoring and tracking application capable of dispelling laptop theft. This research was aimed at investigating a solution to the following question: How to design a middleware that ensures an effective and efficient integration (of biometrics, mobile phones, RFIDs and mobile phones) and for use within the context of asset monitoring and tracking system. Two objectives were identified to help answer the above question: (1) to create a generic middleware architecture that interconnects at least 4 diverse IoT components and (2) to use a laptop monitoring and tracking system (LMTS) to assess the integrity and responsiveness of the middleware.

The rest of this article is organised as follows: Sect. 2 encompasses associated literature, while Sect. 3 elucidates, the methodology used and Sect. 4 details the discussion and conclusion.

2 Allied Literature

2.1 Cyber Physical Systems (CPSs)

According to [1], a cyber-physical system (CPS) connects the cyber and physical worlds for the purpose of merging and analysing real and cyber information – the analysed data is fed back to the real world. Such systems are capable of controlling movement of reality autonomously – without intervention of humans. This characteristic draws parallels with Internet of Things. CPSs intermingle with the physical world by interrogating and actuating. The main sub-systems that make up CPSs are: (1) wireless/wired sensor networks (WSNs); (2) a decision support system (DSS); and (3) physical systems/elements [4]. In their simplest form, CPSs consist of a mobile asset affixed with a sensor node, in a mobile asset system and that navigates around a monitored area [14].

Of interest to this paper is the mobile asset tracking application domain of CPSs. Embedded computing and low power sensing components have driven developments in this area. These systems can either be based on static or mobile networks [7]. Examples of static systems are described in [15, 16]. Most of the applications under the mobile networks category are found in the health sector where they are used to track hospitals valuable assets as well as the patients [17, 18]. They also have been used in tracking cultural artefacts [6].

2.2 Asset Management Systems

From an organization point of view, asset monitoring and tracking is not an isolated exercise; it is rather part and parcel of the organization's asset management system. The system should in particular capture the assets' types, asset life, asset life cycle and asset life stages. The system also needs to be integrated with other organizational functions such as financial management and human resources management [11]. Some of the basic components addressed in this paper are user profile management, asset life/life cycle management and asset monitoring and tracking.

2.3 Underling Platforms for Asset Tracking Systems

The purpose for the establishment of Internet of Things (IoT) was to make our daily endeavours more convenient and affluent by acting as a link between digital and physical worlds [19]. IoT can trace its history at MIT [20] – since then, many definitions of the concept have emerged. The most recent addition is found the Cluster of European Research projects on the Internet of Things (CERP-IoT) [21]. Here the ITU's [22] 4As vision has been extended to 6As. The internet will host approximately 50 billion devices by 2020, according to research commissioned by Cisco [23]. This proliferation of miniature computing and connected devices presents some integration challenges; there are no (known to the authors) mutual standards to support interoperability, interconnection and security of these heterogeneous devices. One of the prevailing difficulty is obscuring the inherent complexity of the environment by safeguarding applications from absolute management of uncongenial network standards, battery-powered tiny inhomogeneous devices that sometimes have constricted computational power, parallelism, data reproduction and fault intolerant networks [24]. The results of poor integration architectures are evident in applications that have scalability, security, interoperability, synchronization and data management issues [25].

The implementation, operation and maintenance of IoT based applications thrive on utilization of middleware services that amongst other things provide a unique platform that conceals hardware heterogeneousness, manage and dissipate commands to sensor nodes, perform data collation, sifting, transportation and storage and considerably boosts the expansion of diverse IoT applications [26].

Radio frequency identifiers (RFIDs) is a smart contact-free technology used to remotely extract data from or transcribe data to an electronic memory chip enveloped within the microelectronic circuit of tags [27]. RFID is designed to remotely and spontaneously identify and locate tagged objects using radio microwaves. Developments in this technology opened avenues to incorporate the technology in a plethora of applications such as remote asset tracking, healthcare and library systems. Although RFID technology is generally cheaper and more stable, it does not support bidirectional communication, which is essential for mobile devices. Moreover, it becomes uneconomical when deployed for tracking relatively smaller assets.

Biometrics entails the systematic examination of biological data and in the context of security, biological data uniquely identifies people by analysing and comparing distinguishing bodily profiles or patterns [28]. The most common biometric security systems deployed encompass fingerprint, palm-print, footprint, facial, iris and voice recognition technologies. Biometric technology is the most reliable, dependable, fool proof, and unobtrusive form of physical identification mechanism albeit expensive.

2.4 Characteristics of Middleware for Internet of Things

A lightweight software layer can realize the practicable operation of an interconnected IoT architecture that enables interoperability and communication between dissimilar or identical IoT objects or a set of sub-layers interposed between the technological and the application levels known as middleware [29]. In [30], a middleware is defined as a software that "supports flexible integration of hardware and application and provides services such as distributed computing environments, remote procedure calls, messaging to users, regardless of the hardware, operating system and network used". Atzori et al. [29] highlighted that middleware has been gaining momentum due to its ability to facilitate development of new services and the interconnection of legacy technologies into new ones, while precluding programmers from understanding diverse technologies implemented at lower layers. The implementation, operation and maintenance of IoT based applications thrive on utilization of middleware services that amongst other things provide a unique platform that: (1) conceals hardware heterogeneousness; (2) coordinate and dissipate commands to sensor nodes; (3) perform data collation, sifting, transportation and storage and (4) considerably boosts the development of diverse IoT applications [26].

An ideal self-sufficient middleware is one that is resilient enough to uphold self-configuration, self-secure, self-optimization and self-attenuating [1]. This guarantees maintenance of the common prominent features of IoT, regardless of the application domain. Some of these features, as explained in [31] are: (1) Ubiquity - which is the state of being everywhere, in some cases concurrently (and explained in the 4As vision of IoT). (2) Affordance - that is, not unsettling the equipoise/environments as professed by users; in other words, the interaction with the users should be as natural as possible and instinctive. (3) Reliable - make certain no trivial disruptions, perpetuity and self-attenuating. (4) Secure - make sure privacy of data is maintained, similar to other conventional systems. (5) Ambient Intelligence (AmI) - that empowers the system to be 'cognizant of' and 'comprehend' situations.

3 Methodology

3.1 Overview

Constructive research approach (CRA) is a methodology whose primary goal is production of novel contemporary knowledge that can be utilized in resolving real world problems, using freshly gained insights and discernments of a phenomenon to rediscover under explored links in pre-existing knowledge. [32]. The selection of CRA for this research was motivated by the need to concoct a tangible novel solution to address laptop larceny at the Central University of Technology, Free State (CUT). Moreover, it was paramount to evaluate the monitoring and tracking system during development, so prototyping was chosen as the development model as it provides opportunities to test the integration IoT devices and Wireless Sensors.

To effectively evaluate both the system prototype and the middleware, experimental research design was used. The research used the following data gathering methods: (1) document analysis - a thorough examination of asset audit reports; and (2) interviews - a face-to-face engagement with head of security along with victims who lost mobile assets to theft ensued. These interviews enabled the researcher to glean key information about theft hotspots, severity and how insecure some buildings are and the vulnerabilities in the currently deployed security systems at CUT. To ensure that every member of the CUT populace has an equal opportunity to be selected as a potential user of the security system under development, purposeful sampling was used to disseminate questionnaires [33]. The purpose of this undertaking was to get a thorough comprehension of the extent of mobile asset theft and to collect users' discernments about the prototype under development.

3.2 Middleware Development

During the development of the prototype, the researcher injected the middleware code into the application; this was a quintessential technique for two reasons: (1) deployment of the system was targeting laptops running Microsoft Windows 7/8 Operating System (MWOS); (2) this operating environment employs various services to manage diverse computing resources such as virtual location sensors, and fingerprint scanners.

The laptop tracking and monitoring middleware borrowed some characteristics from Cougar middleware. To support exchange of data between the centralized database server and connected laptops, the middleware implemented a database interface that uses structured query language (SQL) query commands. The database server was calibrated to process multiple simultaneous queries ranging from data storage, retrieval, updates to deletion. The database interface was developed with resilience in mind and the result was a fault tolerant middleware aimed to use an intelligent algorithm to establish a new connection to a backup database server when connection to the main database server fails.

To support quality of service (QoS) as in MiLAN [34], the tracking and monitoring middleware used mobile phones to facilitate two way communication. This bi-directional communication was achieved by having a laptop through the middleware transmit messages to a mobile phone and the mobile phone in turn requests the laptop through the middleware perform some action such as sending locus data via short message service (SMS) commands. Quality of service was attained through delivery of acknowledgements; that is to say, a confirmation SMS was generated by the laptop and sent through the middleware to acknowledge receipt of service request command to the mobile phone that requested some operation to be performed. Instant generation and dissemination of security breaches through the middleware's SMS service also corroborated the embodiment of quality of service.

The intelligence in IoT applications is realised through their capacity to react to phenomena triggered by diverse parameters or sensory readings. The LMTS middleware designed to listen and respond to several (ephemeral, intervallic and persistent) events spawned by the application, hardware and database layers, middleware services (SMS, location and database). To support parallelism, the middleware treated each event's action as an independent task that the operating system can independently and concurrently execute. This parallel execution of tasks (parallelism involves, modularizing programs into individual components that execute independently on separate threads) [35], resulted in a middleware architecture that is highly responsive and effective in performing preconfigured actions depending on the generated event.

As depicted in Fig. 2 (Adopted from [30]), the middleware design is comprised of three layers that is application, middleware and hardware. Each layer manages and utilizes several modules and services.

(a) Gather Assets Data

The system does not only facilitate the tracking and monitoring of laptops, it is also equipped with the ability to manage high level functions such as registration and storage of laptop information in a database, allocate, re-allocate, transfer or revoke laptop assignment, to or from a university personnel.

(b) Monitoring and Detection

This necessitates autonomous and smart techniques to systematically monitor and detect laptop security breaches. In actuality, this module detects unapproved departure of laptops from the university premises. A blend of RFID readers, RFID tags, cellular phones, Global Positioning System (GPS) devices, Geofencing, alarm bells, wireless sensors, and biometric readers/scanners are used to trigger and broadcast this security violation. To simplify turning on/off the laptop-monitoring task, a fingerprint scanner was used to command the middleware to conditionally start/halt interrogating the tag affixed on the laptop, without setting off a security alarm.

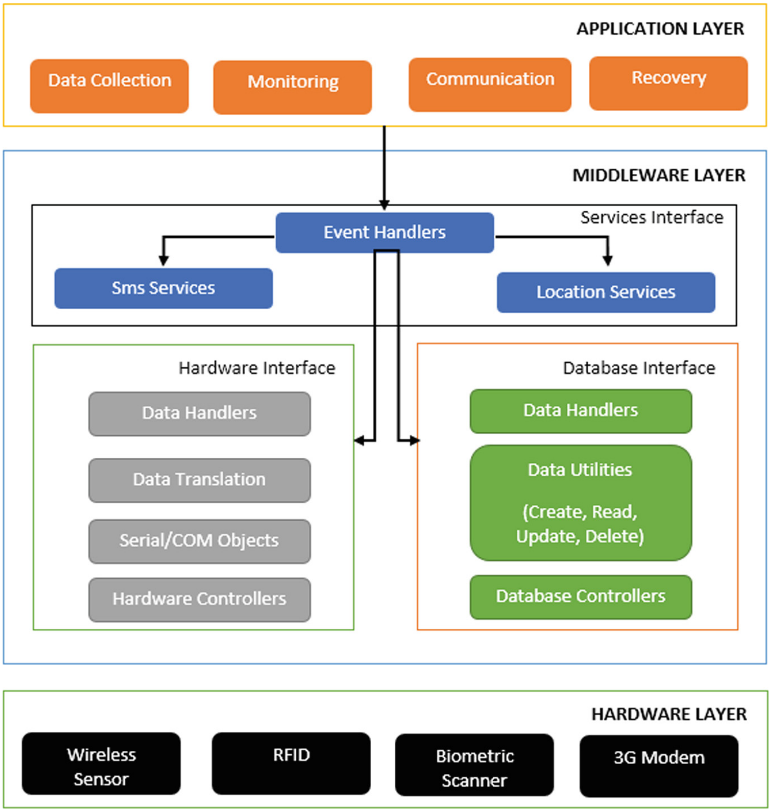


Fig. 2. IoT LMTS middleware architecture adopted from [30]

(c) **Dissemination and Communication**

This component is in charge of transmission of applicable warnings/information concerning to laptop security breaches to all interested parties. It involves triggering alarms; distribution of SMS reports to personnel (security, asset manager in charge of assets at the institution and the person allocated the laptop in question); this prompts the security personnel to take swift action to foil this theft attempt.

(d) **Recovery Capability**

This element entails recovery steps taken in an attempt to track and trace a lost laptop. Despite lacking the much needed intelligence and automation, tracking was achieved through continuous interrogation of the windows virtual GPS sensor to extract the most recent locus data and the same information was delivered via SMS to the victim's mobile phone upon requesting such information using SMS commands. Using locus data periodically saved in the database, Google maps was used to show the physical location of the laptop in question.

3.3 Middleware Services

The following are services and interfaces fulfilled and managed by the middleware:

(a) **SMS Services**

Describes services that manage communication between mobile phones and the LMTS using Ozeki SMS gateway. Ozeki SMS gateway is a software that capacitates computer systems with transmissions of SMS over telecommunication networks using an SQL server database and GPRS modem.

(b) **Location Services**

Involves exploitation of a virtual location sensor or GPS device to harvest physical location data from either windows location data providers or application programming interfaces (APIs) and relay this data to a database server or mobile phone via SMS. The following techniques: (1) global position system (GPS), (2) wireless fidelity (Wi-Fi) triangulation, (3) cell phone tower triangulation, (4) internet protocol (IP address) resolution; are suitable for generation of the physical position of a computer or mobile device [36].

(c) **Hardware Interface**

Entails the mechanism in which electronic peripherals are added to a computer system to expand its capabilities. Standard interfaces for integrating external hardware peripherals with computers systems are serial ports and universal serial bus (USB).

(d) **Database Interface**

Data persistence and access are integral features that a number of applications depend on to perform their respective tasks. An SQL database was designed to enable capturing, storage, retrieval and manipulation of data pertaining to laptop tracking and monitoring. Microsoft's entity framework was used to deliver reliable access to the database stored on Microsoft SQL server. Microsoft SQL Server (is a relational database management system RDBMS).

(e) **Hardware layer**

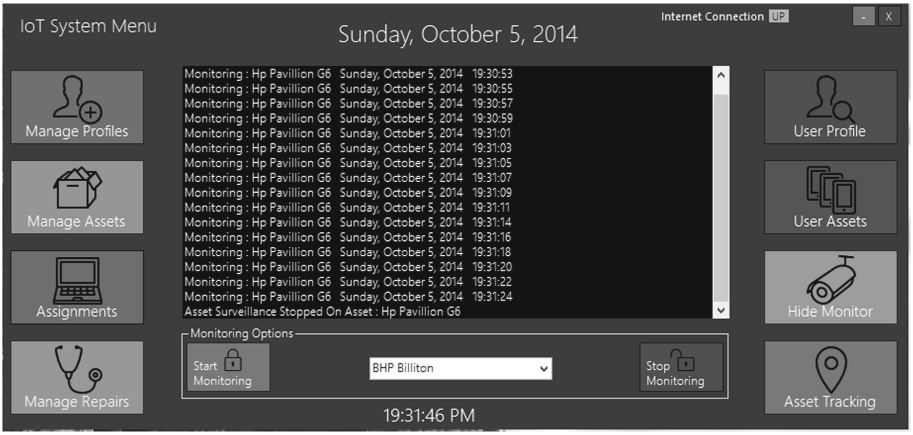
Individual IoT applications are designed to satisfy different organizational needs, as such, they are inclined to interconnect diverse hardware peripherals. Each hardware peripheral offers distinctive data or service and may employ the capabilities of other hardware to transport data reading, this creates coupled systems that depend on other systems to accomplish their designated mandate.

3.4 System Prototype Application and Evaluation

In line with the four main components of the framework shown in Fig. 2, the implementation was as follows:

(a) **Monitoring**

Laptop monitoring is carried out to detect and foil any attempts to steal the asset. This undertaking utilizes a plethora of hardware peripherals along with middleware services. As indicated in Fig. 3, for the LMTS to monitor a laptop, a user must connect either an RFID scanner or pressure/weight sensor to the laptop using a USB cable. From the system's interface, the user must select the building from which the laptop is sheltered and click button "Start Monitoring" to commence monitoring. In response to this button click action, the system requests fingerprint authentication from the asset owner, the user is expected to scan the finger whose fingerprints were captured during asset assignment and stored on the database. The fingerprint comparison process requires access to the database through the middleware's database services. If the fingerprint on the scanner matches a database stored template, the system conducts asset tag validation to verify if it corresponds to the one assigned to the asset during asset registration. This tag validation step is skipped when monitoring is conducted using a pressure/weight sensor. Once the system has authenticated the asset owner and validated the asset tag, laptop monitoring commences; this is conducted through perpetual interrogation of the asset tag or checking for unreasonable fluctuations in laptop weight



```
Monitoring : Hp Pavillion G6   Sunday, October 5, 2014   19:30:53
Monitoring : Hp Pavillion G6   Sunday, October 5, 2014   19:30:55
Monitoring : Hp Pavillion G6   Sunday, October 5, 2014   19:30:57
Monitoring : Hp Pavillion G6   Sunday, October 5, 2014   19:30:59
Monitoring : Hp Pavillion G6   Sunday, October 5, 2014   19:31:01
Monitoring : Hp Pavillion G6   Sunday, October 5, 2014   19:31:03
Monitoring : Hp Pavillion G6   Sunday, October 5, 2014   19:31:05
Monitoring : Hp Pavillion G6   Sunday, October 5, 2014   19:31:07
Monitoring : Hp Pavillion G6   Sunday, October 5, 2014   19:31:09
Monitoring : Hp Pavillion G6   Sunday, October 5, 2014   19:31:11
Monitoring : Hp Pavillion G6   Sunday, October 5, 2014   19:31:14
Monitoring : Hp Pavillion G6   Sunday, October 5, 2014   19:31:16
Monitoring : Hp Pavillion G6   Sunday, October 5, 2014   19:31:18
Monitoring : Hp Pavillion G6   Sunday, October 5, 2014   19:31:20
Monitoring : Hp Pavillion G6   Sunday, October 5, 2014   19:31:22
Monitoring : Hp Pavillion G6   Sunday, October 5, 2014   19:31:24
Asset Surveillance Stopped On Asset : Hp Pavillion G6
```

BHP Billiton

Fig. 3. IoT LMTS monitoring interface

and compare it against the threshold. By clicking “Stop Monitoring” button, the owner requests the system to stop monitoring the asset weight or reading the asset tag without triggering a security breach. In responds to this request, the system requests fingerprint authentication from the laptop owner just to verify if the request is coming from the person who initiated the monitoring process.

(b) **Transmission and distribution of alerts**

This component operates through utilization of Ozeki SMS gateway (As of June 2017, Ozeki website), GPRS modem, SQL database, middleware services and mobile phones. This module is responsible for the transmission of related warnings or information to relevant personnel (asset owner, security guards and asset manager) via SMS messages upon detection of a laptop security breach.

(c) **Laptop Recovery**

This module entails recovery steps taken in an attempt to track and trace a lost laptop. Despite lacking the much needed intelligence and automation, tracking was achieved through continuous interrogation of the windows virtual GPS sensor to extract the most recent locus data and the same information was delivered via SMS to the victim’s mobile phone upon requesting such information using SMS commands. Using locus data periodically extracted from the virtual sensor and saved in the database, Google maps was used to display the physical location of the laptop in question. Microsoft windows operating system has APIs or dynamic link libraries (DLLs) that make available the location service interface to any application that intends to query locus data from the native code layer. From this code layer, it is possible to calibrate the accuracy level of the virtual sensor to meet the requirements of the host application (As of June 2017, Microsoft MSDN website).

4 Discussion and Conclusion

This paper presented both the system prototype and the embedded laptop monitoring and tracking middleware architecture. The proposed middleware implemented a variety of service components such as: (1) locus data extraction from windows virtual location or GPS sensor; (2) using RFID reader and passive tags to conduct laptop surveillance; (3) two-way transmission of SMS messages; and (4) utilization of database services to facilitate data management through SQL commands.

The adoption of common characteristics found in revered middleware solutions such as MiLAN and Cougar transcended the LMTS middleware architecture into a hybrid middleware that is cost-effective versatile, fault tolerant, reactive, and suitable for the mushrooming Cyber Physical Systems (CPSs). The middleware demonstrated versatility through its support for parallel processing of triggered events and bi-directional communication without compromising the quality of monitoring and other supported services. This quality of being versatile was also displayed by the middleware’s ability to use windows operating system services and resources to manage an array of hardware peripherals such as the GPRS modem, RFID and fingerprint scanner. The presented system prototype demonstrated conformance to the objectives set for this case study.

The LMTS middleware in this study was constructed using concepts proposed by Hwang and Yoe [30] and around the IoT paradigm and the outcome thereof was a middleware architecture suitable for use in tracking and monitoring systems within the CPS and IoT paradigms. The middleware provided a proficient and flexible interface to interact with heterogeneous IoT hardware peripherals, trigger events based on generated data, manage, process and consume data generated by diverse interconnected devices. The core purpose of this middleware was to create a standardized environment to manage diverse hardware by concealing their heterogeneity. The evaluation of the middleware and the LMTS was conducted by observing how reactive the system was to diverse events, and measuring the time, it took to detect and broadcast security violations. Other tests comprised of discerning data returned by SQL query commands, the accuracy of locus data generated by the virtual sensor and lastly the ability to correctly interpret SMS commands and trigger the correct action.

Constructive research approach (CRA) was adopted in this research study and a real-life solution was concocted following a seven-step process mooted in [37]. These steps are: (1) identification of real life problem; (2) a thorough investigation of the proposed panacea with respect to long term objectives; (3) an in-depth examination of the problem domain; (4) translation of requirements into system artefacts such as use cases, flow-charts and data-flow diagrams (DFDs); (5) develop a working prototype using designs created in the previous step; (6) prototype deployment, testing and evaluation; and (7) discussion and conclusion.

References

1. Liu, X.F., Shahriar, M.K., Al Sunny, S.M.N., Leu, M.C., Hu, L.: Cyber-physical manufacturing cloud: architecture, virtualization, communication, and testbed. J. Manuf. Syst. **43**, 352–364 (2017)
2. Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M.: Internet of Things (IoT): a vision, architectural elements, and future directions. Future Gener. Comput. Syst. **29**, 1645–1660 (2013)
3. Seo, A., Jeong, J., Kim, Y.: Cyber physical systems for user reliability measurements in a sharing economy environment. Sensors **17**(8), 1868 (2017)
4. Zhou, P., Zuo, D., Hou, K.M., Zhang, Z.: A decentralized compositional framework for dependable decision process in self-managed cyber physical systems. Sensors **17**(11), 2580 (2017)
5. Gunes, V., Peter, S., Givargis, T., Vahid, F.: a survey on concepts, applications, and challenges in cyber-physical systems. KSII Trans. Internet Inf. Syst. (TIIS) **8**, 4242–4268 (2014)
6. Rodriguez-Sanchez, M.C., Borromeo, S., Hernández-Tamames, J.A.: Wireless sensor networks for conservation and monitoring cultural assets. IEEE Sens. J. **11**, 1382–1389 (2011)
7. Kim, K., Jin, J.Y., Jin, S.I.: Classification between failed nodes and left nodes in mobile asset tracking systems. Sensors **16**(2), 240 (2016)
8. Liu, H., Darabi, H., Banerjee, P., Liu, J.: Survey of wireless indoor positioning techniques and systems. IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.) **37**(6), 1067–1080 (2007)

9. Kim, T.H., Jo, H.G., Lee, J.S., Kang, S.J.: A mobile asset tracking system architecture under mobile-stationary co-existing WSNs. Sensors **12**(12), 17446–17462 (2012)
10. Chen, Z., Xia, F., Huang, T., Fanyu, B., Wang, H.: A localization method for the internet of things. J. Supercomput. **63**(3), 657–674 (2013)
11. ISO: Asset Management – Management Systems Requirements (2014). http://www.iso.org/, http://www.iso.org/iso/iso-55089-colour_pdf.pdf
12. Balakrishnan, D., Nayak, A.: An efficient approach for mobile asset tracking using contexts. IEEE Trans. Parallel Distrib. Syst. **23**, 211–218 (2012)
13. Schadler, T., Yates, S., Wang, N., Sharma, A.: Mobile workforce adoption trends. Forrester Research (2013)
14. Kim, K., Chung, C.W.: In/Out status monitoring in mobile asset tracking with wireless sensor networks. Sensors **10**(4), 2709–2730 (2010)
15. Giannoulis, S., Koulamas, C., Emmanouilidis, C., Pistofidis, P., Karampatzakis, D.: Wireless sensor network technologies for condition monitoring of industrial assets. In: Emmanouilidis, C., Taisch, M., Kiritsis, D. (eds.) APMS 2012. IAICT, vol. 398, pp. 33–40. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-40361-3_5
16. Rajendran, N., Kamal, P., Nayak, D., Rabara, S.A.: WATS-SN: a wireless asset tracking system using sensor networks. In: 2005 IEEE International Conference on Personal Wireless Communications. ICPWC 2005, pp. 237–243. IEEE, January 2005
17. Jeong, S.Y., Jo, H.G., Kang, S.J.: Fully distributed monitoring architecture supporting multiple trackees and trackers in indoor mobile asset management application. Sensors **14** (3), 5702–5724 (2014)
18. Balakrishnan, D., Nayak, A.: An efficient approach for mobile asset tracking using contexts. IEEE Trans. Parallel Distrib. Syst. **23**(2), 211–218 (2012)
19. Gershenfeld, N., Krikorian, R., Cohen, D.: The internet of things. Sci. Am. **291**(4), 76 (2004)
20. Gama, K., Touseaui, L., Donsez, D.: Combining heterogeneous service technologies for building an internet of things middleware. Comput. Commun. **35**(4), 405–417 (2012)
21. Jain, A.K., Hong, L., Pankanti, S.: Internet of Things - strategic research roadmap, Technical report, Cluster of European Research projects on the Internet of Things, September 2009. http://www.internet-of-things-research.eu/pdf/IoTClusterStrategicResearchAgenda2009.pdf
22. International Telecommunication Union: ITU Internet Report 2005: The Internet of Things. International Telecommunication Union, Geneva (2005)
23. Evans, D.: The internet of things: how the next evolution of the internet is changing everything. CISCO white paper, vol. 1, pp. 1–11 (2011)
24. Han, S.W., Yoon, Y.B., Youn, H.Y., Cho, W.-D.: A new middleware architecture for ubiquitous computing environment. In: 2004 Proceedings Second IEEE Workshop on Software Technologies for Future Embedded and Ubiquitous Systems, pp. 117–121. IEEE (2004)
25. Hadim, S., Mohamed, N.: Middleware: middleware challenges and approaches for wireless sensor networks. IEEE Distrib. Syst. Online **7**(3), 1 (2006)
26. Römer, K., Kasten, O., Mattern, F.: Middleware challenges for wireless sensor networks. ACM SIGMOBILE Mob. Comput. Commun. Rev. **6**(4), 59–61 (2002)
27. Finkenzeller, K.: R.F.I.D Handbook. Wiley, Chichester (2003)
28. Jain, A.K., Kumar, A.: Biometrics of next generation: an overview. Second Gener. Biom. **12** (1), 2–3 (2010)
29. Atzori, L., Iera, A., Morabito, G.: The internet of things: a survey. Comput. Netw. **54**(15), 2787–2805 (2010)
30. Hwang, J., Yoe, H.: Study on the context-aware middleware for ubiquitous greenhouses using wireless sensor networks. Sensors **11**(5), 4539–4561 (2011)

31. Rodríguez-Molina, J., Martínez, J.-F., Castillejo, P., López, L.: Combining wireless sensor networks and semantic middleware for an internet of things-based sportsman/woman monitoring application. Sensors **13**(2), 1787–1835 (2013)
32. Crnkovic, G.D.: Constructive research and info-computational knowledge generation. In: Magnani, L., Carnielli, W., Pizzi, C. (eds.) Model-Based Reasoning in Science and Technology. SCI, vol. 314, pp. 359–380. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15223-8_20
33. Teddlie, C., Yu, F.: Mixed methods sampling a typology with examples. J. Mixed Methods Res. **1**(1), 77–100 (2007)
34. Heinzelman, W.B., Murphy, A.L., Carvalho, H.S., Perillo, M.A.: Middleware to support sensor network applications. IEEE Netw. **18**(1), 6–14 (2004)
35. Silberschatz, A., Galvin, P., Gagne, G.: Applied operating system concepts. Wiley, Hoboken (2001)
36. Doty, N., Mulligan, D.K., Wilde, E.: Privacy issues of the W3C Geolocation API. arXiv preprint arXiv:1003.1775 (2010)
37. Kasanen, E., Lukka, K., Siitonen, A.: The constructive approach in management accounting research. J. Manag. Account. Res. **5**, 243 (1993)