

Improving Memory Exploitation in M-health Applications through Context-Aware System

Siham Belhadi^{1,*} and Rachid Merzougui¹

¹Department of Telecommunication, Tlemcen University, STIC Laboratory, Tlemcen, Algeria

Abstract

INTRODUCTION: The mobile application requires some information about its environment in order to the adaptation of its behaviour according to this information in a transparent manner. However, such a task usually needs the resource consumption of mobile devices.

OBJECTIVES: the main goal of this work was the design of the context aware infrastructure for the healthcare applications embedded on mobile devices. This last aims to the integration of the context awareness into the execution environment and providing a adequate methodology of the generic context manager, with the maintenance of low level the resource consumption.

METHODS: This collected the context information (medical data) thanks to the component (Context Element Monitor). When the components set acquire values of different elements context which belong the same class, these components run in the same time. it is mean the execution of its in a group way. From the state of monitors (ACTIVE or DISABLED), our system creates the new context situation. The mobile application is structurally an assembly of services. The adaptation methodology revolves around the execution manager of services according to the new created context situation. This approach ensures the best use of resources depending on the characterization of context and reduces the amount of contextual information in the adaptation processes.

RESULTS: We evaluated a memory space in three cases, since it is the most resource that is affected in the field of the mobile health (mobile patient monitoring application). For this, we estimated the consumed memory at two levels: the first is the management module "Context Manager" and the second level considers the adaptation procedure. In the Context Manager, the first and the third case were the exploited memory space was almost identical, Knowing that its consumption is varied between 1556 and 2026 KB. But in the second case, it increased and become varying between 2424 and 6848 KB. the other hand, in the adaptation mechanism, it is necessary a huge space of a memory for managing services more than 40000 kb.

CONCLUSION: Our own approach of context manager and adaptation can open the new way in the m-health applications development. This allows these applications that can exploit the great quantity and the different type of a medical data with the best use of the memory space.

Keywords: Context Awareness, Resource Consumption, Mobile Health.

Received on 14 January 2021, accepted on 28 January 2021, published on 07 April 2021

Copyright © 2021 Siham Belhadi and Rachid Merzougui, licensed to EAI. This is an open access article distributed under the terms of the [Creative Commons Attribution license](https://creativecommons.org/licenses/by/4.0/), which permits unlimited use, distribution and reproduction in any medium so long as the original work is properly cited.

doi: 10.4108/eai.7-4-2021.169177

*Corresponding author. Email: belhadi.ttl@gmail.com

1. Introduction

Nowadays, most people use their Smartphone in their daily activities such as health, education, transport, etc. In this

range, these applications can use in different domains by its installation on Smartphone. Unfortunately, this has had a negative effect on the capacity and the speed of the mobile device for running these applications. However, unquestionably, the success of these applications resides in

the best exploitation of limited resources and providing needed services to mobile users.

This paper is particularly focused on the topic of the mobile health (M-health) which becomes an important sub-segment of the field of electronic health (e-Health) [1]. Different applications of the mobile health can create, collect, store and also transmit information in order to ensure the safety of both the patient and the doctor [2]. Nevertheless, these applications can consume the further resource. In this scope, it has to use a reasonable amount in terms of the used resource.

Concurrently, the impressive progression of programming world over the last years has achieved a qualitative step through the use of the context awareness in mobile software applications. This concept is defined by [3], as a system which is context-aware if it uses context to provide relevant information and/or services to the user, where relevancy depends on the user's task. Whereas context-aware systems usually need the processes set and the different techniques of an information management. This information collects from varying sources. The objective of these systems is the adaptation of the mobile applications behavior and improving the application quality to its external require.

The number of M-health applications is expanding, but some of them do not achieve the success due to the resource consumption in form reasonable (a great quantity and a various quality of the medical information) [4], [5]. For this, our work aims to the proposal of the system, which manage the execution of applications. Therefore, it ensures the best use of the resource. For this goal, we assume inserting the context-aware in M-health applications through the system that allows guaranteeing the right management of a medical information, as well as providing appropriated services to the mobile user with a guarantee of the perfect exploitation of the limited resource.

The rest of this paper is structured as follows: Section 2 presents some related works. Sections 3 introduces the M-health applications. Section 4 define the context in the field of the mobile health. Section 5 and 6 describe the context-aware health service (CAHS) and the operation of our framework. we present the simulation environment in Sect. 7. We study and analyze the obtained results in Sect. 8. We conclude in Sect. 9.

2. Related works

A large number of researchers have published their work that related to a topic of the mobile health, but the issue of the resource consumption stays as the challenge. In this dimension, we take into account this major problem in this paper.

The resource consumption relies on the functionalities and the architectural design [6] being exploited. Whereas a mobile application uses the different functionalities, where the appropriated architectural design is essential for this application. Currently, mobile applications depend on context-aware system for developing and facilitating their

functionalities through the flexible and simple architectural design. The scope of context-aware systems can follow the different architectural styles: a server-centric [7] [8], a mobile-centric [9] [11] or a hybrid architectural style [12].

The construction design of the context-aware system is based on the layered conceptual architecture, so that each layer presents the certain functionality, relevant challenges, such as context acquisition, retrieval of raw context data, preprocessing, storage and Management as well as Application layer [13]. Unfortunately, these tasks have had a negative effect on the limited resource. In this particular, K. P. Subbu and all [14] indicate that local versus centralized treating and energy consumption are still the important challenges. Whereas [5] are focused on mobile applications. Although this is a major step that should be studied in any context-aware system, such as mentioned in [15] [16], it do not exist a generic analysis which related on the resource consumption.

Especially when focusing on the resource consumption for managing mobile applications, in [17], authors propose a framework which manages the contextual information of mobile applications, dedicated to the transport domain. In addition, they measure a quantity of the used memory in a different process that this framework is performed. In [18], authors evaluate and discuss several architectural styles of context-aware mobile applications, as well as, provide a guidelines set to decide the right architecture for a particular application depending on its characteristics. The use of the appropriated architectural style strongly improves the exploited resource in the context-aware mobile applications execution. Other work [19] presents a new precise Context-Aware Memory Profiling (CAMP) framework that efficiently traces all the memory dependences with full context information. CAMP statically analyzes a context tree of a program that illustrates all possible dynamic context and simplifies context management during profiling.

M-health and various associated topics have witnessed the big development in the very last years, but it has been lacking and needing the management of its mobile services for the best use of the limited resource in the different scenarios. Our work focuses on the integration of context awareness in the management of M-health services. We assume the management approach which based on the context group and creating the new context situation. This is what differentiates our proposal from other existing ones. Therefore, this paper poses a study and offers a methodology to facilitate the various medical data management and to provide the different services with improving the resource use in M-health applications execution. This research work estimates the exploited memory space for managing the contextual information and adapting the mobile services.

3. M-health applications

The development of the mobile computing and the communication technology is taking an important place in the health field. It is worth mentioning Vital Wave

Consulting, a research and consulting firm [20], which defines the mobile health as "using mobile communications, such as PDAs and mobile phones, for health services and information". The survey [2] presents the various mobile health applications that are intended to help patients and health professionals.

Throughout the day, people exploits their mobile terminals (a telephone, a laptop) in different ways, but most of these users need several services which allow to help in their health life, including monitoring medical data (blood sugar, ECG, EEG, etc.), health training (medication, food, symptoms of disease, etc.) and other information about nearby entities (pharmacy, medical center, hospital, etc.). These terminals add the different advantages to this domain, despite there are many constraints of these terminals in terms of the processing capacity of medical data, rates of the application installation and execution as well as the battery life. In addition, this field is distinguished by the qualitative and the quantitative data, making it more complication in terms of the medical data management and the service execution.

In this reason, our proposal articulates around the context-aware system which is characterized the simple strategy of the context management and the execution manager for facilitating the operation service. The performance evaluation is based on two pathologies (cardiac and diabetes). We chose these pathologies because the processes of its medical data treatment are different.

4. Context in the field of mobile health

The important factor for the creation of a context-aware system is namely the context.



Figure 1. Context in computing system

With regard to the context, authors in [21] suggested several definitions for the concept of context. However, the most general definition, which was assumed by [3], which

defines the context as "Any information that can be used to characterize the situation of the entity that can be a person, place, or object". The context is the circumstance of the interaction between users, applications, and the surrounding environment. In addition, the context considers as a set of elements (CE_i). These elements represent all the aspects that are susceptible to influence the mobile service execution in the healthcare domain. These elements identify through the particular situation of the user, as mentioned in [22].

In this scope, it expresses the set of medical and user data, which allows helping in the remote patient monitoring. if the contextual information has several common tasks, there are the existence of the relation between them. For example, in the case of a diabetes, a change in the activity of patient (walking, sleeping) may affect in level of his vital signs (blood glucose).

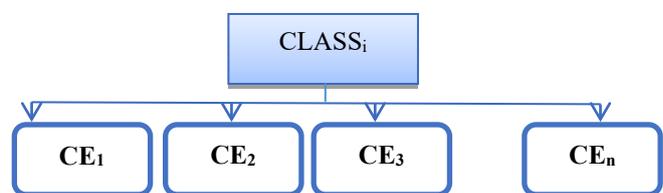


Figure 2. Context diagram

Figure 2 depicts the proposed context diagram which uses in the possesses of the context acquisition and the context situation creation. On the one hand, this classification facilitates the use of a medical information for detecting the health situation of the mobile patient, because it is more simple and clarity. On the other, it eases to create the context situation, which exploited in the management of services. This methodology doesn't need the great quantity of an information in the execution manager processes. This improves the performance and reduces the resource consumption (exploited memory space).

the goal of M-health applications is the medical situation monitoring of patients. In this scope, we assume that $Class_i$ is the health environment, which presents the medical information (Figure 3). It becomes an essential class for monitoring the medical situation. Critical elements revolve around the health environment data: vital signs, prescribed medication and activity.

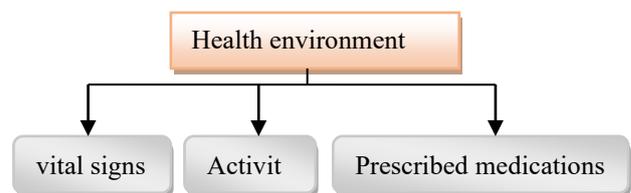


Figure 3. Context diagram for the class is health environment

5. Context-Aware Health Service (CAHS)

In our past work, we proposed the context-aware framework (CAHS) which is based the service oriented architecture (SOA). Our model is composed of a set of key components such as the management modules and mobile services. These components and the completed architecture functionality are described in [22], whose details are out of the scope of this paper.

In [22], iPOJO and android is particularized for an implementation and a validation; as a result, we obtained CAHS, a system which manages the execution of applications dedicated in M-health. Its high-level architectural design, illustrated in Figure 4, is composed of three modules of the management, namely context manager, adaptor, SDC controller, which continuously operate as long as the framework is functioning.

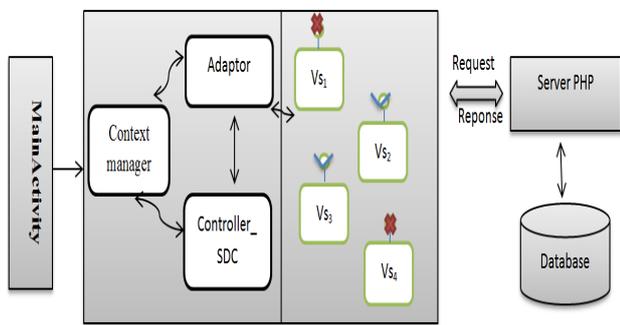


Figure 4. CAHS architecture

The main advantage is modules of the common management, allowing applications to be lighter and to exploit the reasonable space of the memory. The architecture of the CAHS Framework constitutes of the management modules the Context manager provides the treatment and analysis processes related to the context information; the Controller-SDC creates the new context situation depending on this information and the adaptor perform the execution of the appropriated services to this situation.

The Android application is implemented to start the execution of the Apache felix-ipojo and the manager modules. It provides two main functionalities: (1) acquiring the context elements values and creating the context situation depending on these elements and (2) the adaptation of services according to this new situation. The presented paper focuses on (1) and (2), because these tasks normally need the big amount of the resource.

6. Operation of our framework

The CAHS framework provides an execution environment that allows the health services to adapt according to context situations at during the execution, without any explicit request from the user. it aims to manage the set of

applications that monitor the several patients suffering from different chronic diseases.

6.1. Collection of context elements values and creation the context situation

With the said applications, several data can be needed for analyzing and detecting the medical situation of the patient. These applications obtain the context information (medical data) through the component, namely Context Element Monitor (M_{CEi}). We propose the management strategy of this monitor for improving the resource use. Our proposal depends on the execution of monitors in a group way.

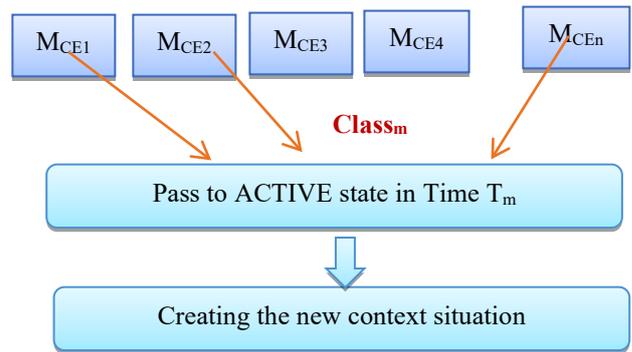


Figure 5. Processes of our system

Figure 5 explains the processes that our system passes from the context information collection until the context situation creation. All monitors of the context elements which belong to the same class, pass to ACTIVE state in the time T_1 . From the state of monitors (ACTIVE or DISABLED), our system creates the new context situation, noted by:

$$Sc_j = \{CE_{1j}, CE_{2j}, \dots, CE_{nj}\} \quad j=1, \dots, m.$$

The approach of the context situation creation adds several advantage for our model. Among them is following:

- The simplification of the complicated context information.
- The creation of the most semantic and reasonable contextual situation.
- The simplicity and clarity of the methodology for exploiting the relationships between elements.
-

6.2. Adaptation of services according to this context situation

Our proposal focuses on applications made up of multiple functionalities. Each functionality presents a certain service. This last can introduce many versions of implementation depending on the current context (Figure 6). Our suggestion is similar to Heralut's proposal [23], such that his idea revolves around technical services with several personalities as well as the suggestion of SPL product families [24].

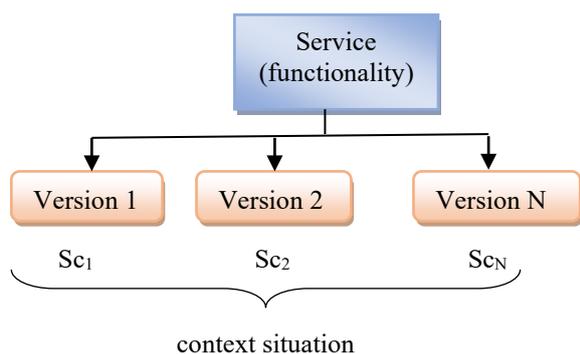


Figure 6. Service with its versions

In our work, the application is structurally presented in the form of an assembly of services. The adaptation methodology revolves around the execution manager of services according to the context situation. Our idea approximates to the suggestion of M. Moeiz [25], which assume an assembly of services in the adaptation approach. In this axe, we have relied on the concept of context-dependent services (CDS) [17]. The said notion outlines the relation between services versions and the context elements. Our proposal bases on the Context Situation Dependent Service (ScDS), as showed in the figure (Figure 7).

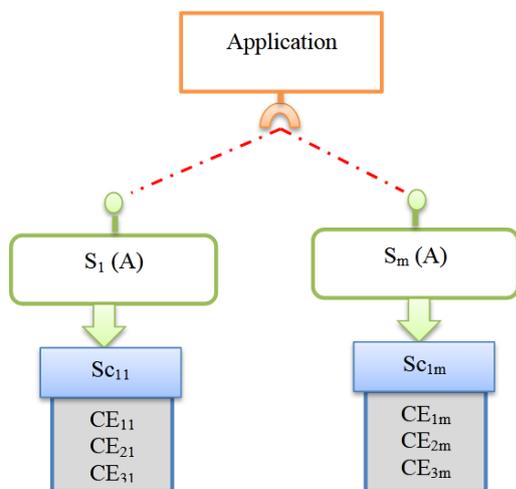


Figure 7. Service with its versions

Our goal is to assemble applications dynamically and autonomously during runtime, allowing to improve the adaptation process. This last based to select the appropriated versions of services. for this, it is necessary to define a description of these versions in order to ensure the correct and consistent assembly of applications. This description (Figure 8) is considered as the identification information. The purpose of it is to optimize the adaptation process where the choice of version is quickly, easily and correctly.

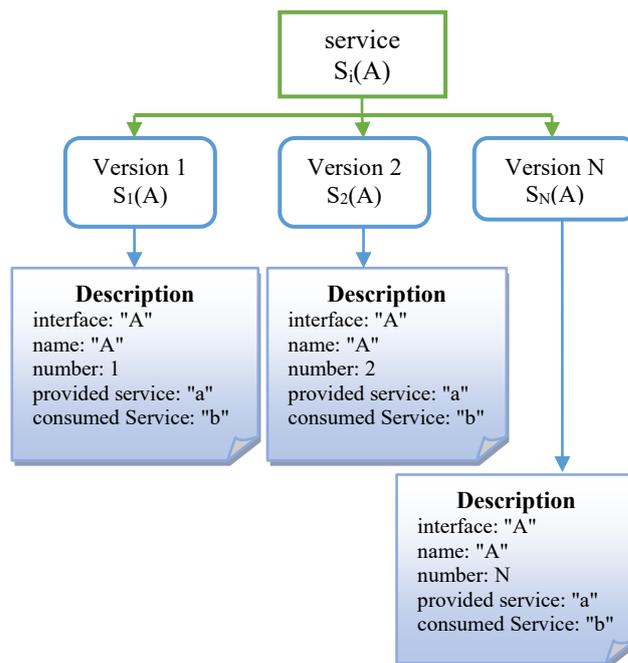


Figure 8. Description of versions Service

Our adaptation process is performed thanks to conditions, that specify through clearly explicit rules. It is achieved by the following condition:

If $k = j$ then the version $S_k(T)$ starts as a function of Sc_{ij} .

The objective of it resides in running of the needed service version according to a number of the context situation; in other words, the number of the version and the situation must be the same. This version is specified by it description, which is presented by the metadata file. This one finds at the level of each version, as represented in Figure 9. The proposed methodology allows acquiring the great quantity and the different quality of an information in an organized way as well as adapting the different services depending to the least amount of context information. This last optimizes the use of the limited resource. As part of our work, we study and analyze our proposal in terms of the exploited memory space.

```

Bundle-ManifestVersion: 2
Bundle-Name: S_affiche1
Bundle-RequiredExecutionEnvironment: JavaSE-1.7
Bundle-SymbolicName: S_affiche1
Bundle-Version: 1.0.0.qualifier
Created-By: iPOJO 1.12.1
Import-Package: android.app, android.graphics, android.view, android.w
idget, com.inter.service.analyse, inter.web.service, org.apache.felix
.ipoj;version=0.8.0, org.apache.felix.ipoj.architecture;version=0.8
.0, org.example.Saffiche, org.inter.serclient, org.osgi.service.cm;ve
rsion=1.2, org.osgi.service.log;version=1.3
iPOJO-Components: instance { $component="com.service.affiche.Saff.Serv
ice_affiche" }component
    
```

Figure 9. Manifest.mf of a version 1 of a display service

7. Simulation environment

We have evaluated our proposal on Android Smartphone. We chose to implement this model using the OSGi technology and the iPOJO component model. Let us describe the functionalities of these technologies as following:

Android is an open-source operation system, which is developed by Google and the Open Handset Alliance [26]. It has a Java virtual machine, named Dalvik JVM. This last doesn't run as the usual (.class). The Android SDK compiles files, which carry the extension (.dex). The language of the development is Java, but the JVM is specific to the system Android [27]. There are various devices (tablets, Smartphone and TV setup) in the market, that use this mobile operation system "android platform" [26].

The OSGi™ platform standard [28] determines a standardized, components-oriented computing environment for networked services. It is the basis of an enhanced Service-Oriented Architecture (SOA). It ensures the security for deploying extensible and downloadable java-based service applications, named bundles. OSGi bundle defines applications collected in a standard Java Archive (JAR) file, and contains a manifest file, a series of compiled Java classes, and native code. A manifest file is the description of the relationships among bundles and the OSGi framework. This one provides an execution environment which dynamically changes the state of bundles (installation, updates, and uninstalling) without restarting the system.

iPOJO is a prototype of a service-oriented component Framework implemented on the top of OSGi™. It is relied on POJO's (plain old Java object). It facilities to inject non-functional code into the application components running, if it is necessary. iPOJO components consist of a component implementation that is managed by a reusable container. IPOJO containers provide the relations of the POJO to the "external" world. IPOJO component uses the annotations or the metadata.xml file to specify the information, including module dependency, provided/required services, configuration. The main advantage of OSGi resides in that providers and consumers no longer need to register/lookup services explicitly. the containers are to do it for them [29]. Indeed, iPOJO containers are not monolithic container, but are composed by handlers which separately developed from the iPOJO core framework. This allows developers to had non-functional properties [30].

The Android application (We used the version 4.4) is a host for our platform. This application is built of a class set, allowing to design our CAHS prototype. The purpose of the Android application is to trigger OSGi platforms, the iPOJO model and the CAHS management modules as well as the M-health applications that run on CAHS. In addition, Android allows monitoring and evaluating the execution of experiments.

Figure 10 illustrates the incompatibility challenge of iPOJO component jar files with the Dalvik virtual machine, knowing that Android uses Dalvik. To do this, it is necessary to extract the classes.dex file for each iPOJO

bundle. To achieve this goal, we use a jar2dex folder, by running the following command:

```
d2j-jar2dex class.jar -o classes.dex
```

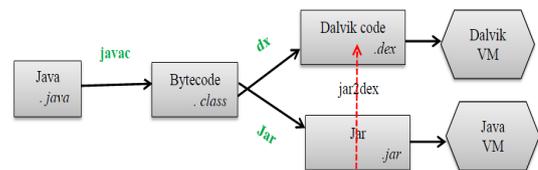


Figure 10. Compatibility of OSGi with Android

We interest the field of telemedicine that has various characteristics, including the different users (patient or doctor), the heterogeneity of medical information (ECG signal, blood sugar, EEG signal) and the great amount of data. For this reason, we have carried out a depth study of our presented system, taking into account all possible probabilities. At this level, we distinguish three users because our research focuses on two diseases (diabetic, heart disease). For this, we distinguish three cases in this evaluation:

Case 1: The user is a patient who suffers from diabete. the monitoring application dedicated to diabete. This application uses a set of the service, that are: S= { S (display), S (analysis), S (connection)}. These services depend on the following context elements:

CE1 {Name = Blood glucose level; Type = Discrete; Class = Health environment}.

CE3 {Name = Activity; Type = Discrete; Class = Health environment}.

CE4 {Name = Prescribed drug; Type = Discrete; Class = Health environment}.

CE5 {Name = Memory; Type = Discrete; Class = Device}.

CE6 {Name = Battery; Type = Discrete; Class = Device}.

Case 2: The user is a patient with a heart disease. The application used in this case exploits the following services: S= { S (display), S (analysis), S (connection)}. Here are the elements used by these services:

CE2 {Name = ECG; Type = Continuous; Class = Health environment}.

CE3 {Name = Activity; Type = Discrete; Class = Health environment}.

CE4 {Name = Prescribed drug; Type = Discrete; Class = Health environment}.

CE5 {Name = Memory; Type = Discrete; Class = Device}.

CE6 {Name = Battery; Type = Discrete; Class = Device}.

Case 3: The user is a doctor which uses the mobile application composed of two services : S= { S (display), S (connection). These services need the ensuing context elements:

CE5 {Name = Memory; Type = Discrete; Class = Device}.

CE6 {Name = Battery; Type = Discrete; Class = Device}.

In this scope, we focus on the evaluation the resource consumption (memory space) when the context management

and the adaptation performed. For this reason, we consider two steps for measuring the used memory space when the context elements values have been collected by the Context Monitor and wish to see how memory it exploits until the appropriated service is ready for use. The results we introduce are the averages and the distributions of values for the (20-30) trials.

8. Result and evaluation

In this study, we focus on a memory space since it is the most resource that is affected in the field of the mobile health (mobile patient monitoring application). for better to analyze the performance of our proposal, we estimated the consumed memory at an operating level of the management module "Context Manager". The results of this section are presented in the following figure (Figure 11).

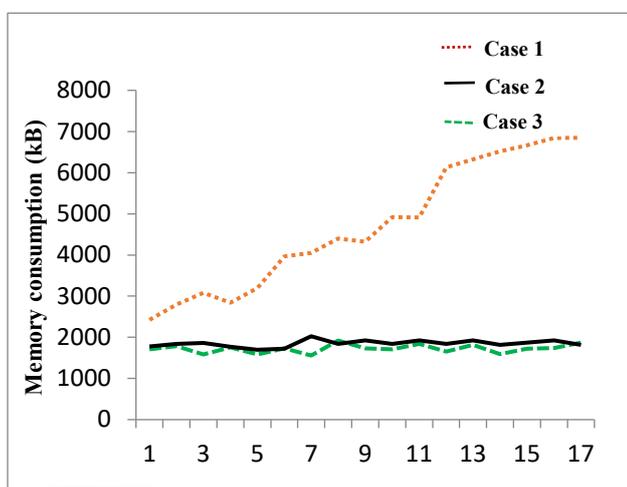


Figure 11. used amount of the memory in the context manager processes

In the first and the third case, we can see that the exploited memory space is almost identical, knowing that its consumption is varied between 1556 and 2026 KB. In contrary, the consumed amount of the memory increases in the second case, so that it value varies between 2424 and 6848 KB. As a result, we might conclude that the memory consumption depends on the type of context elements, making of necessary the identification of the data amount, when a context type is "Continuous", without any effect on the detection of the patient's current medical situation.

For a perfectionist evaluation of the effectiveness of our CHAS platform in terms of the resource consumption, we measure the consumed memory at an adaptation of applications after the processes of the context management. The figure (Figure 12) shows the evolution of the used memory in the three cases (see the simulation environment).

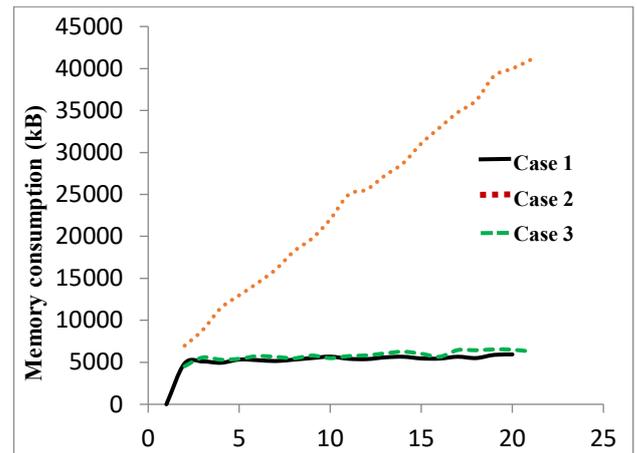


Figure 12. A used amount of the memory in the adaption processes

In the figure (Figure 12), the adaptation mechanism needs a huge space of a memory for managing services, more than 40000 kb. This capacity exponentially increases in case 2. Nevertheless, we observe the stability of the memory consumption, as well as slight variations in others cases (first and third). Therefore, we can conclude that there is a clear impact of the context type on the memory consumption.

The results of this evaluation may be due to the virtual machine, which does not have the needed moment to free the memory between two successive tests. Therefore, we take into consideration this aspect so that the adaptation operations perform on the long time intervals.

For the validation, our evaluation and the work results [31] were compared. In [31], the amount of the used memory in the services manager was considered the sum of the exploited memory space in test 1A, test 1B and test 1C. It is easy to deduce that the memory consumption of CAHS varies between 3000 kb and 40000 kb in the three cases for managing the M-health services; however, for CATS, it varies between 12552 kb and 52870 kb.

Through this comparison, we demonstrate that our proposal has achieved the best performance and improved the resource consumption (memory space).

9. Conclusion

In this paper, we consider a context-aware system and propose an approach for improving the resource consumption. Aiming at challenging scenarios with different and high quantity of the medical data, the proposed manager methodology effectively exploits the relation between the context elements and the context situations to manage the context and services. Based on the SOA technique, our framework architecture is the modular assembly, which allows to reduce the components that are operated. We build our platform to evaluate the performance of the proposed solution. The simulation results show that the suggested approach can provide the advantage in terms of

the best use of the memory space, permitting to exploit the different type of a medical data in the mobile health application. As result, our model is capable of monitoring various diseases. Note that in order to better perfect the performance of our system, it is highly required to identify the number of values, when the type of context elements is continuous. In addition, the case of monitoring the patient who suffers from several diseases was not considered. Thus, in our future work, we depend on work [18] for including these different challenges in the CAHS framework.

Acknowledgements.

We thank committee of International Symposium on Advanced Electrical and Communication Technologies for accepting our paper. Moreover, we would like to thank also reviewers for carefully reading our work.

References

- [1] S. Akter, and P. Ray, "mHealth-an ultimate platform to serve the unserved," MIA Yearbook of Medical Informatics, pp. 75-81, 2010.
- [2] B. M C Silva, J. J P C Rodrigues, I. de la Torre Díez, M. López-Coronado, and K. Saleem, "Mobile-health: A review of current state in 2015," Journal of Biomedical Informatics, vol. 56, pp. 265-272, Aug. 2015.
- [3] Abowd G.D., Dey A.K., Brown P.J., Davies N., Smith M., Steggles P. (1999) Towards a Better Understanding of Context and Context-Awareness. In: Gellersen HW. (eds) Handheld and Ubiquitous Computing. HUC 1999. Lecture Notes in Computer Science, vol 1707. Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-48157-5_29
- [4] H. Khalid, E. Shihab, M. Nagappan, and A. E. Hassan, "What do mobile app users complain about?," IEEE Softw. vol. 32, pp. 70-77, 2015.
- [5] N. Capurso, B. Mei, T. Song, X. Cheng, and J. Yu, "A survey on key fields of context awareness for mobile devices," J. Netw. Comput. Appl, vol. 118, pp. 44-60, Sep. 2018.
- [6] J. Berrocal, J. Garcia-Alonso, C. Vicente-Chicote, J. Hernández, T. Mikkonen, C. Canal, and J. M. Murillo, "Early analysis of resource consumption patterns in mobile applications," Pervasive Mobile Comput, vol. 35, pp.32-50, Feb. 2017.
- [7] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, "A survey of mobile cloud computing: Architecture, applications, and approaches," Wireless Commun. Mobile Comput. vol. 13, pp. 1587-1611, Dec. 2013.
- [8] D. S. Yadav, and K. Doke, "Mobile cloud computing issues and solution framework," Int. Res. J. Eng. Technol. 3 (11), pp. 1115-1118, Nov. 2016.
- [9] J. Guillen, J. Miranda, J. Berrocal, J. Garcia-Alonso, J. M. Murillo, and C. Canal, "People as a service: A mobile-centric model for providing collective sociological problems," IEEE Softw. vol. 31, pp. 48-53, Mar.-Apr. 2014.
- [10] S. Tramp, P. Frischmuth, T. Ermilov, S. Shekarpour, and S. Auer, "An architecture of a distributed semantic social," network. Semantic Web. vol. 5, pp.77-95, 2014.
- [11] F. S. Tsai, W. Han, J. Xu, and H. C. Chua, "Design and development of a mobile peer-to-peer social networking application," Expert Syst. Appl. vol. 36, pp.11077-11087, Oct. 2009.
- [12] R. Morrow, "Hybrid application design: Balancing cloud-based and edge based mobile data," Gigaom, Austin, TX, USA, Tech. Rep, NOV .2013.
- [13] M. Baldauf, S. Dustdar, and F. Rosenber, "A survey on context-aware systems," Int. J. Ad Hoc and Ubiquitous Computing, vol. 2, pp. 263-277, 2007.
- [14] K. P. Subbu., and A. V. Vasilakos, "Big data for context aware computing, Perspectives and challenges," Big Data Res, vol. 10, pp. 33-43 Dec. 2017.
- [15] B. Djoudi, C. Bouanaka, and N. Zeghib, "A formal framework for context-aware systems specification and verification," J. Syst. Softw. vol. 122, pp. 445-462, Dec. 2016.
- [16] Truong HL., Juszczak L., Manzoor A., Dustdar S, "ESCAPE – An Adaptive Framework for Managing and Providing Context Information in Emergency Situations", In: Kortuem G., Finney J., Lea R., Sundramoorthy V. (eds) Smart Sensing and Context. EuroSSC 2007. Lecture Notes in Computer Science, vol 4793. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-75696-5_13
- [17] D. Popovici, M. Desertot, S. Lecomte, and T. Delot, "A framework for mobile and context-aware applications applied to vehicular social networks," Social Network Analysis and Mining. vol. 3, pp. 329-340, 2013.
- [18] G. Ortiz, A. García-De-Prado, J. Berrocal, and J. Hernández. "Improving Resource Consumption in Context-Aware Mobile Applications Through Alternative Architectural Styles." IEEE Access. vol. 7, pp. 65228 – 65250, 2019.
- [19] Ch. Kim, J. Kim, J. Kang, J. W. Lee, and H. Kim. "Context-Aware Memory Profiling for Speculative Parallelism." in Proc. 24th International Conference on High Performance Computing (HiPC), 18-21 Dec. 2017, Jaipur, India, IEEE, 2018.
- [20] Vital Wave Consulting, "mHealth for Development: The Opportunity of Mobile Technology for Healthcare in the Developing World," Washington, D.C. and Berkshire, UK: UN Foundation-Vodafone Foundation Partnership, 2009.
- [21] X. Li, M. Eckert, José-F. Martínez, and G. Rubio, "Context Aware Middleware Architectures: Survey and Challenges," sensors. vol. 15, pp. 20570-20607, Aug. 2015.
- [22] S. Belhadi, and R. Merzougui, "Design and implementation of a context-aware health service platform (CAHS)," International Journal of Electrical and Computer Engineering. vol. 9, pp. 4993-5005, Dec. 2019.
- [23] Colombe Herault, Adaptability of technical services in the components model, PhD Thesis, University of Valenciennes and Hainaut-Cambresis, French., 2005.
- [24] Carlos Parra, Xavier Blanc, and Laurence Duchien, Context awareness for dynamic service-oriented product lines, 13th International Software ProductLine Conference SPLC2009 (John McGregor and Dirk Muthig, eds.), vol. 1, 08 2009, pp. 131-140
- [25] M. Miraoui, "Software Architecture for pervasive computing: context modeling and dynamic adaptation of services," Thesis of PhD. Montreal: Ecole de technologie supérieure, 2009.
- [26] L. Kumar, and K. Sharma, "Web Based Novel Technique for Watermarking Colour Images on Android Mobile Phones," International Journal of Advanced Research in Computer Science and Software Engineering. vol. 3, pp. 47-52, July. 2013.
- [27] S. Ghouali, "Développement d'applications de Télémedecine sur Smartphones," Thesis of PhD. Tlemcen: University Abou Bekr Belkaid, 2017.
- [28] M. Chen , J. Chen , and T. Chang , Android/OSGi-based vehicular network management system, in Proc. The 12th

International Conference on Advanced Communication Technology, 7-10 Feb. 2010, Phoenix Park, South Korea, IEEE, 2011, pp.169-183.

- [29] D. Popovici ,M. Desertot , S. Lecomte ,and T. Delot," A framework for mobile and context-aware applications applied to vehicular social networks," Social Network Analysis and Mining, vol. 3, pp. 330-340, 2012.
- [30] C. Escoffier, R S. Hall, and P. Lalanda," iPOJO: an Extensible Service-Oriented Component Framework," in Proc. IEEE International Conference on Services Computing, 9-13 July 2007, Salt Lake City, UT, USA, IEEE, 2007.
- [31] P. Dana," Context management for mobile applications dedicated to transport," Thesis of PhD, University of Valenciennes and Hainaut-Cambresi, French, 2012.