RESEARCH ARTICLE OPEN ACCESS

A Mobile-Cloud based Context-Aware and Interactive Framework for Diabetes Management

Asmae Benali*, Bouchra El Asri**
**(IMS Team, SIME Laboratory ENSIAS, Mohamed V University, Rabat, Morocco*
*** (IMS Team, SIME Laboratory ENSIAS, Mohamed V University, Rabat, Morocco*

ABSTRACT
One of the biggest preoccupations of any healthcare provider is trying to eliminate the mistakes during treatment. Using Cloud computing permits to host all information in one place and make it accessible anywhere, anytime, and any channel, especially when it comes to the disease diabetes mellitus. Diabetes mellitus is a group of diseases characterized by an elevated blood glucose level (hyperglycemia) resulting from defects in insulin secretion, in insulin action, or both. It is, today, the most challenging syndrome in the world. In the latest survey, the world's 65% of the population is suffering from either Type 1 or Type 2 diabetes mellitus. The patient's blood glucose level is not the same 24x7 hours in most of the cases and take medication 24x7 hours is impossible. Cloud Computing is so the best solution to check in the patient's blood glucose control and try to balance it, especially at remote areas where healthcare services aren't easily available.
Keywords: Context-Aware, Cloud Computing, Diabetes, Health Care, Software Product Lines

I. INTRODUCTION

Cloud computing has recently become a prevalent technology and considered as one of the main trends in the information and technology sector. It designates a model on which computing, storage, and software is provided as a service which is accessible from anywhere in the world on demand. Healthcare is considered as one of the most important sectors in all countries, where developed countries expended an average of 9.5% of their domestic product as based on 2012 statistics cited in [1]. Minimizing the occurrence of mistakes during treatment is one of the major concerns of any healthcare provider, especially when it is about critical diseases that need a permanent and daily monitoring like diabetes. In fact, most of diabetics suffer from the difficulty of managing their treatments, as this disease requires a regular supervision; watching the blood glucose level, taking medicaments, regularly visiting their physician, keeping their records and performing medical checkups.

The rest of this paper is organized as follows. Section II presents the background of our work. In Section III, we compare and position our solution with other proposals found in the literature. Section IV describes our proposed solution. Finally, in section V, we conclude and introduce our future work.

II. BACKGROUND

In this section, we introduce a comprehensive background on the diabetes diseases and its data management systems. In addition, we present a review on the classical and mobile-based applications for diabetes management.

2.1 Diabetes Mellitus

Diabetes happens because the body can't use glucose correctly either the pancreas not producing enough insulin or the cells of the body not reacting rightly to the insulin produced. This disease has three main types; type I, type II and Gestational diabetes. People living with diabetes may have to cope with short-term or long-term complications as a result of their condition and many other associated health problems. But maintaining a healthy lifestyle, Control of blood pressure, attending regular check-ups and monitoring glucose levels can avoid or minimize these problems [2]. Diabetes affects at least 285 million people in the world as reported by the International Diabetes Federation [3], and that number estimated to increase and reach 438 million by the year 2030, with two-thirds of all diabetes cases happening in low- to middle-income countries. Indeed, Countries with limited infrastructures for diabetes care have a high percentage of Diabetics as a result of the bad management of this epidemic. A daily Management of Diabetes is a condition to keep blood pressure and cholesterol levels under control, maintaining weight and it will consequently help to reduce the risk of developing complications.

2.2 Cloud computing and Diabetes Mellitus

Cloud Computing is a model that permits to enable ubiquitous and on-demand access to a shared collective of configurable computing resources (e.g., storage, computer networks, servers, applications

and services) which can be quickly provisioned and easily managed. Cloud computing provide its clients with diverse capabilities to store and process their data in data centers.

The bad management of diabetes is due globally to the diabetes data stored in different places. When all information of diabetic person is hosting in one place and it is accessible anywhere, the patient, the physician, and the caregiver can have a complete view of everything that impacts the blood glucose levels and subsequently can contribute to a significant and durable improvement in blood glucose control. The access and the management of diabetes information over internet can ameliorate the engagement and help diabetic persons to stay more connected to their self-care especially the people who live in remote areas where healthcare assistance is not easily available.

2.3 Diabetes Mobile-based Management Systems

With the evolution of technology, Management of diabetes has become easy and uncomplicated. A home blood glucose test was a major leap ahead for diabetics, after the patient tests his blood glucose levels, the management system records and stores the test results and other detailed data like, date and time of tests, the type and dosage of insulin, type of exercise, diet, etc...

The data acquired can then be presented as a chart form in order to permits the patient to clearly follow the progressing of their blood glucose management.

2.4 Software Product Lines

The product lines constitute an approach that has already been used successfully in many fields, permitting stakeholders to optimize the product development by adopting a systematic reuse of common elements that are shared by several products throughout the software development process. It allows the development of a set of software products that satisfy the product requirements and assure a major gain in resource, quality, and time [4]. SPL is a set of systems that share a group of manageable features, as depicted in [5], where a feature is defined as an end-user visible characteristic of the system [6] and developed from a set of base assets. An asset, in the context of SPL, is a software artifact which can be employed in the development of several products. An asset can be a model, software component, a document, or any other unit that contribute in the development of system. Software Product Line Engineering (SPLE) is divided on two main processes; domain engineering and application engineering. The domain engineering refers in general to the creation of assets that will be used in the product line, whereas the application engineering takes charge of the construction of the final products with particular requirements.

2.5 Variability in SPL

Variability modeling is considered the permitting technology for delivering in a fast, comprehensive and consistent way a large variety of software systems of high quality. The principle is to identify and build reusable assets that can be used to construct new products reducing the effort, the cost and the time invested when building new products from scratch. Pohl et al. [7] introduce the notions of variability subjects and objects to describe variability. Variability subjects present the variant properties from the real world, whereas a variability object defines a way in which variability subject changes. Variability in software product lines can refer to essential and technical variability [8], external and internal variability [9], product line and software variability [10]. In software product line engineering, a variation point is a representation of a variability subject and a variant is the equivalent term for variability objects which identifies a single option of a variation point.

2.5.1 Feature Diagrams

The notion of feature diagrams was first proposed by Kang et al. in 1990 [6], as a tool in the Feature Oriented Domain Analysis (FODA). The feature diagram serves as way to model variability among a set of identical products. The commonalities and variabilities among products in the same domain are presented as features. A feature is a prominent or distinctive and user visible aspect, quality, or characteristic of a software system or systems [11]. A feature can be either [12] i) Mandatory, it exists in all products if the parent is selected; ii) Optional, it is not identified in all products; iii) Alternative (One Of), it specializes more general feature; only one, option can be chosen from a set of features iv) Or: One or more features may be selected in the product.

III. RELATED WORKS

Nowadays, mobile phone has become the main sources of information for users. In fact, a huge number of applications were developed in different mobile operating systems to respond to the user's requirements. Many applications have been created to manage diabetes.

BGluMon (Blood Glucose Monitor) is a mobile application that permits the patient to see clearly his/her blood glucose level on daily basis; it is downloadable on Apple Store [13].

Glucose Buddy is an application in Apple Store, which allows the management of patient's logs such as, blood glucose level, weight, medication, food and exercise. The results are

presented in the form of grid table where analysis is missing [14].

WaveSense application contains a meter that can be connected to any Apple device so as to measure blood glucose level through strips defined for this meter. When the measurement of blood glucose is done, the data is automatically recorded in WaveSense application [15]. However, this meter is available only in some countries; so the use of this application is limited.

Vree is a mobile application for Apple iOS which is specified for type II diabetes. This application has many functions and calculations, but these later can not be used for the other types of diabetes [16].

In [17], On Track is a mobile application in Android Marker to manage diabetes by tracking level of glucose in the blood, food, medication, weight. By tracking these values, the application can present a summary about the patient's progress and save it. But this application lacks of some functionalities like connectivity.

IV. PROPOSED FRAMEWORK

In this section, we present a context-aware framework that combines mobile computing and cloud computing with software product lines variability techniques in order to provide a smart healthcare system for diabetes.

The mobile application, due to the well developed smart phones, is destined to be a fast and efficient method to connect among the system, the patient, the medical care institutions, and the emergency units. Data can be vastly and robustly obtained through a communication channel at low expenses along the country economizing time and resources. Moreover, the instruction guide is yielded to the user anytime and anywhere through the same channel.

Besides, diabetes medical sensors are combined with the mobile in order to acquire continuously feedbacks and fresh updated medical record from the patients. Because of its availability, scalability, and huge storage capabilities, cloud computing is included to accommodate the system so as to support the huge number of users that try to access on the run.

4.1 System Architecture

The proposed framework is based on mobile cloud computing and software product lines approach to handle the variability of both the patient context and the change of the cloud service environment. The framework architecture is depicted in the Fig. 1 below.

4.1.1 Functional Requirements Analysis

At first, we identify the actors. Then we decline their different behaviors and interactions with the system. The actors and users of the system are showed in Fig. 2:

- **Diabetic Patient:** patient provides and stores through the mobile application their basic information about its profile; include the name, address, and gender, date of birth, the personal medical record, and other genetic records. Patient can also provide all information related to his disease management like daily blood glucose value, medications, measurement scenario, regular check-ups, and Contact persons in case of emergency. In addition, patient can establish and follow his diet by introducing the daily information of the food consumed and exercise. On the other hand, patient will be able to view and print their medical records.

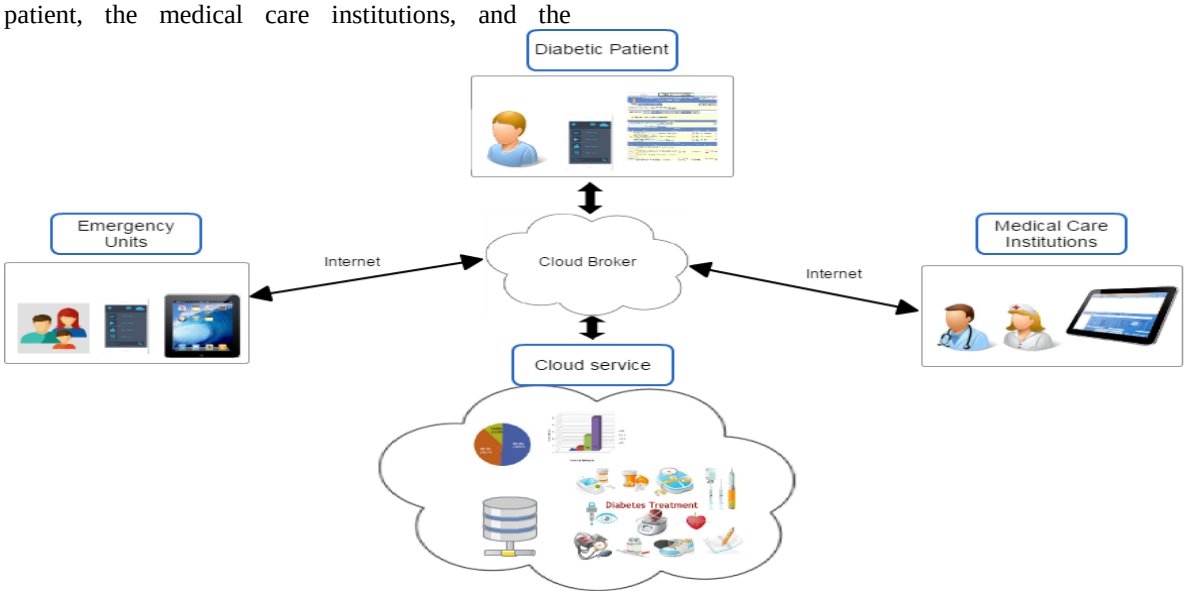

Fig. 1. Our system architecture.

- **Medical Care Institutions:** can be doctors, nurses, Workers at laboratories, radiology centers, that maintain and manage personal medical record inserted by patient, modify it with medical information and then store it as electronic health record in the cloud and shared it with professionals specialized in Diabetes. In addition, Diabetes professionals can import additional data in electronic health record; as laboratory tests and x-ray photos.
- **Emergency Units:** can give a real-time warning to a caregiver family about patient's measurement and send them the location signal of the patient.

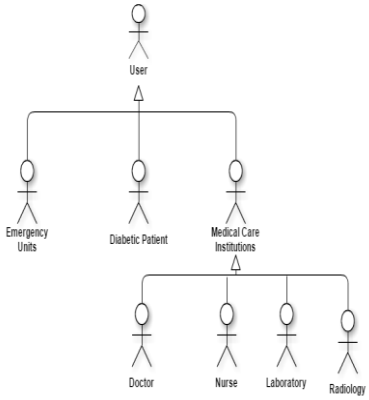

Fig. 2. Actors of the System.

Use cases are a very useful and effective means of capturing the functional requirements. Fig. 3 below presents the use case diagram of our system which expresses the different interactions of the actors with the system.

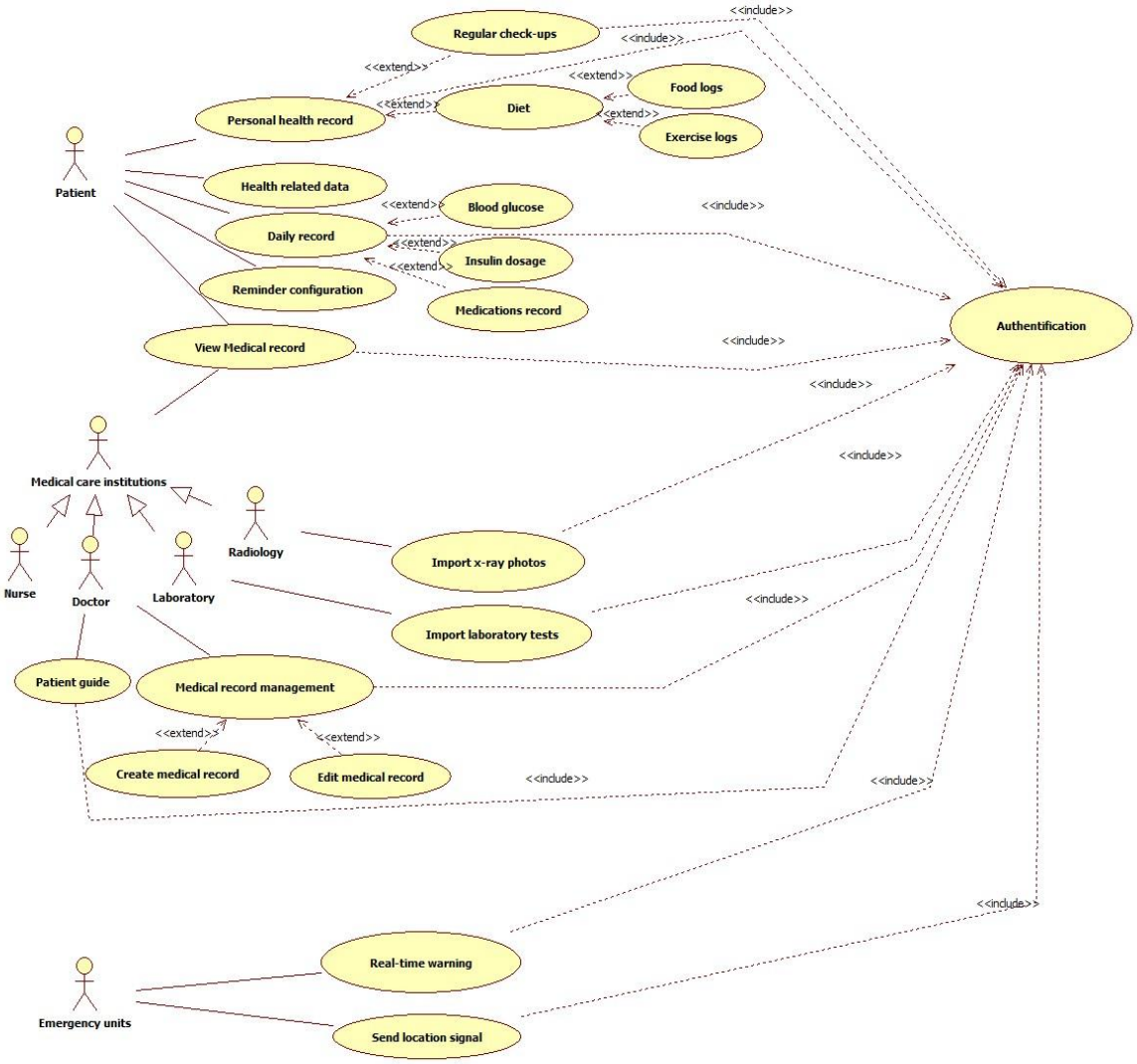

Fig. 3. System Use Case Diagram.

The system collects real-time and fresh information about the patient and its environment such as blood glucose values, measurements scenario, times, location etc… and uploads it to the cloud broker using GPRS protocol. Consequently, the system will illustrate the received information as a feature model and generates the corresponding XML-format (see Fig. 4)

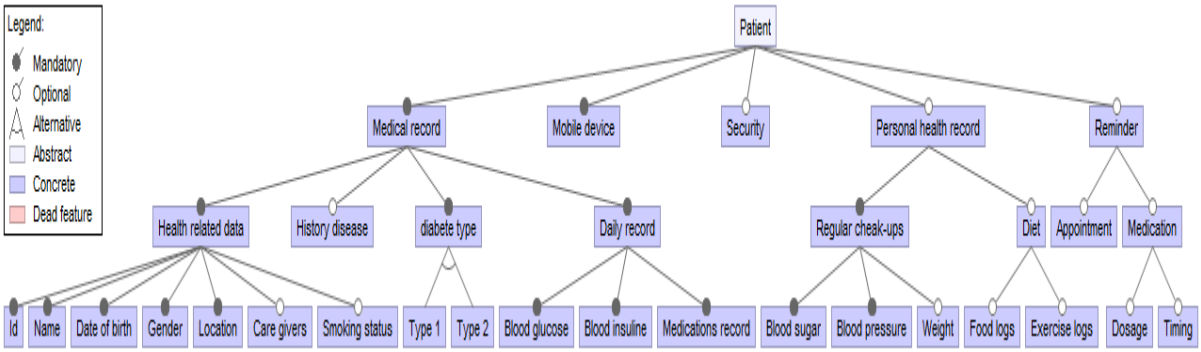

Fig. 4. Diabetic patient - feature model.

The system will analyze the patient information to conclude the health status of the patient and further service information that is required by the application like the quality of the service. According to this information, the service provided will be presented as a feature model and select then the features which will be included in the provided service (see Fig. 5).

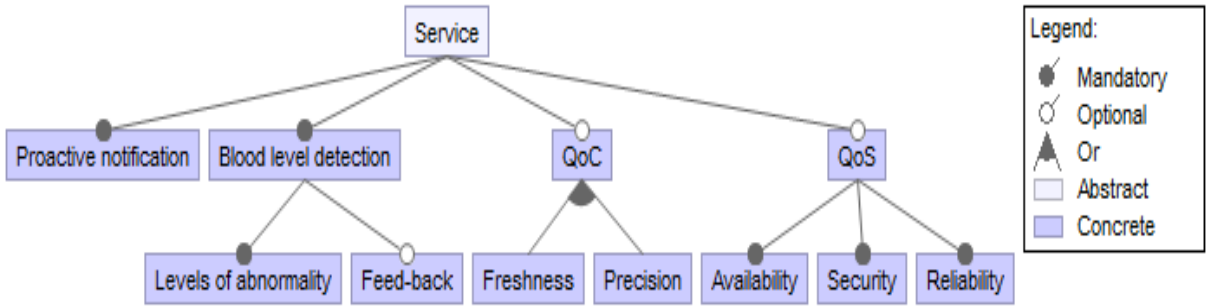

Fig. 5. Cloud service- feature model

In addition, according to the measurements scenario and basic information of the patient, the system will determine the blood level as abnormality level or as a feed-back as presented in Fig. 6.

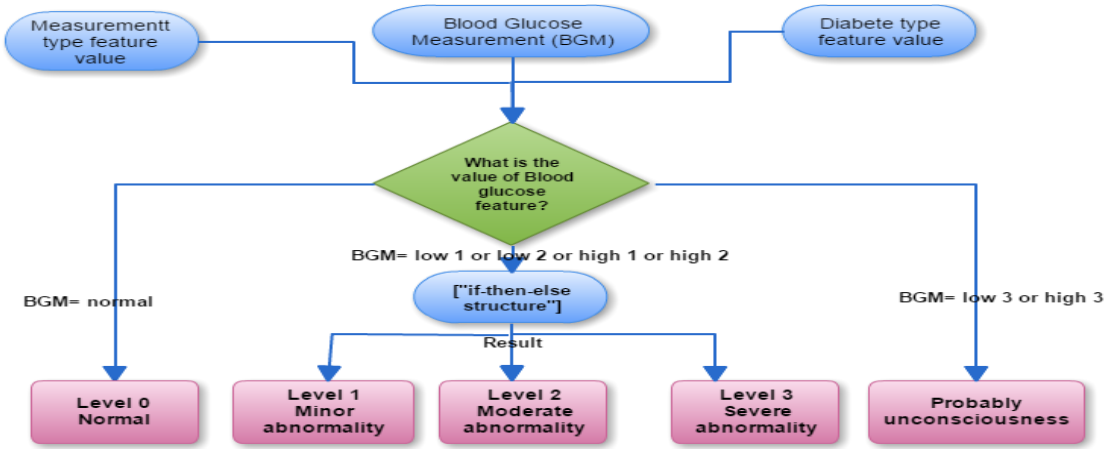

Fig. 6. The process of the blood glucose abnormality.

When the patient health status is critical, the system will automatically send a notification warning to the patient's caregivers and send at the same time the patient location signal to the emergency units so as

as to save the patient life.

We created a number of rules and procedures, according to discussions with doctors and related references published by the World Health Organization [18], for abnormal blood glucose detection.

Blood-glucose abnormalities are separated into five levels: level 0 signifies a normal state of the diabetic patient; while level 1 implies a minor blood-glucose abnormality which demands observation. Level 2 presents a moderate abnormality, with no impending danger, that needs urgent hospitalization. According to Level 3, it indicates an upcoming danger which may be accompanied by risk of coma or unconsciousness. Level 4 signifies the most dangerous scenario, where the patient may be, at any time, unconscious.

The system produces different actions based on the abnormality level. When the cloud broker received the patient's blood glucose data, it automatically uploads the related data to the Blood level detection service, for example blood-glucose values, measurement scenario and time. This service determines the levels of abnormality and the feed-back to be provided.

4.1.2 Non Functional Requirements

E-Health systems enable improved overall access to services and Health information. To safely manage and share access to information in a sensible and critical system like health system, it is indispensable to be able authenticate all users that interact with the system, including people and organizations. This section presents the recommended non functional requirements of our system that improve the services offered by such systems like security, resilience, quality of service, performance and quality of context.

The requirements of the system are numerous.

Among them, it should be noted:

- **Confidentiality:** The system must ensure the confidentiality of the data.
- **Safety:** The identity of the patient and other data transmitted in the Networks must be guaranteed through the use of secure network systems.
- **Authorization:** The access rights to the application must be specified (patient, medical corps, responsible ...).
- **Interoperability:** It should guarantee a high degree of interoperability between systems that exchange information and data among themselves.
- **Data compatibility:** The system must use digital formats and structures to enable information in circulation to be exchanged between systems developed by different manufacturers.
- **Hardware compatibility:** It must ensure a high degree of compatibility between different applications such as medical devices and digital systems in order to be able to exchange for example diagnostic images.
- **Economic Efficiency:** The system must be designed in such a way as to Longevity in time. To this end, it must be prevented from rapidly becoming obsolete due to the introduction of new standards for e-health.
- **Availability:** Information must be updated in real time to ensure better traceability and better responsiveness from hospitals and doctors.
- **Resource Utilization:** The resources and servers used must be robust, possess high storage capacity and use good backup techniques.
- **Response Time:** It is important to have real-time access to patient medical data, especially when it concerns emergency situations. Therefore, the response time of the "Medical Record" functionality should not exceed few seconds.
- **Reliability:** Reliability is a non functional requirement that is hard to pin down especially when is about an E-health system. In fact, our system muss predicts, analyzes, prevents and mitigates failures over time. For instance, make a store more reliable by normalizing its data, to remove redundant copies, work with reliable protocols (TCP) and unreliable protocols (UDP), convince customers of a disk's reliability by advertising an MTBF (Mean Time Between Failures) of a million hours, or a long warranty period, or even a mysterious technology like SMART (Self-Monitoring, Analysis and Reporting Technology).

- **Quality of Context:**

Context-awareness is a key requirement for applications in mobile and pervasive environments. Dey and Abowd [19] defined Context-Aware as: "A system 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". Context-awareness systems aim to adapt their behavior to the dominant situation in the environment by sensing and using context information [20].

Quality of context is coined to indicate quality of context information. According to Buchholz et al. [20] have defined the QoC as: Quality of Context (QoC) is any information that

describes the quality of information that is used as context information. Thus, QoC refers to information and not to the process or the hardware component that possibly provide the information. These indicators may be the precision, freshness, probability of correctness, temporal resolution and spatial resolution [20][21].

V. CONCLUSION

In this paper, we proposed a smart healthcare framework for diabetes that combines smart phones and cloud computing technologies. It is based largely on software product line approach to handle the variability of patient's context such as location, measurements scenario, daily record and also to take into account the environment change of the services generated by the provider, like the blood level detection that depend on the patient's information received, availability of the resources, security, etc…

In ongoing work, we plan to develop interactive mobile applications that cover other endemic diseases basing on our proposed framework.

REFERENCES

[1] Organization of economics co-operation and development: Available from http://www.oecd.org/ , Accessed 2 November 2016.

[2] Diabetes: Available from https://www.diabetes.org.uk/Guide-to diabetes/Complications/, Accessed 1 November 2016.

[3] International Diabetes Federation. IDF Diabetes Atlas. Epidemiology and Mobility: Available from http://www.idf.org/. Accessed on 5 November 2016.

[4] P. Clements, and L. Northrop, "Software Product Lines: Practices and Patterns", Addison-Wesley Longman Publishing Co., Inc., 2001.

[5] Software Engineering Institute. A Framework for Software Product Line Practice, 2010.

[6] Kyo C. Kang, Sholom G. Cohen, James A. Hess, William E. Novak, and A. Spencer Peterson. Feature-oriented domain analysis (foda) feasibility study. Technical report, Carnegie-Mellon University Software Engineering Institute, November 1990.

[7] Klaus Pohl, Günter Böckle, and Frank J. van der Linden. Software Product Line Engineering: Foundations, Principles and Techniques. Springer-Verlag, 2005.

[8] G. Halmans, and K. Pohl, "Communicating the variability of a software product family to consumers", Software and Systems Modeling, 2(1):15–36, 2003.

[9] K. Pohl, G. Böckle, andF. J. van der Linden, "Software Product Line Engineering: Foundations, Principles and Techniques", Springer-Verlag New York.

[10] A. Metzger, K. Pohl, P. Heymans, P. Schobbens, and G. Saval, "Disambiguating the Documentation of Variability in Software Product Lines: A Separation of Concerns, Formalization and Automated Analysis",

[11] K.C. Kang, S.G. Cohen, J.A. Hess, W.E. Novak, and A.S. Peterson,"Feature-oriented domain analysis (FODA) feasibility study", DTIC Document,1990.

[12] J.V. Gurp, J. Bosch, and M. Svahnberg, "On the notion of variability in software product lines" in Proceedings of the Working IEEE/IFIP Conference on Software Architecture, pp. 45, 2001.

[13] "bglumon,". Available from :http://itunes.apple.com/us/app/bglumonblood-glucosemonitor/id353118307?mt=8, Accessed 30 November 2016.

[14] Glucose buddy. Available: http://www.glucosebuddy.com/. Accessed 30 November 2016.

[15] IBG Star: Available from: http://www.ibgstar.us/default.aspx. Accessed 30 November 2016.

[16] "Vree," Apple Store: Available from http://itunes.apple.com/us/app/vree-fordiabetes/id355923059?mt=8. Accessed 30 November 2016.

[17] "On Track," Android Market: Available from:https://play.google.com/store/apps/details?id=com.gexperts.ontrack&hl=en .Accessed 30 November 2016.

[18] "Diagnosis and Classification of Diabetes Mellitus," Diabetes Care, vol. 34, no. 1, 2011, pp. S62–S69; doi: 10.2337/dc11-S062.

[19] A. Dey and G. Abowd. "Towards a better understanding of context and context-awareness", in CHI 2000 Workshop on The What, Who, Where,When, and How of Context-Awareness, 2000.

[20] T. Buchholz, A. Kpper, M. Schiffers."Quality of context: What it is and why we need it?", In Proc. of the 10th International Workshop of the HP OpenView University association (HPOVUA), 2003.

[21] K. Sheikh, M. Wegdam, and M. Van Sinderen. "Quality-of-Context and its use for Protecting Privacy in Context Aware Systems", Journal of Software, vol. 3(3) pp. 83-93, March 2008.