

Diabetes Monitoring System Using Mobile Computing Technologies

Mashaal Saud Bin-Sabbar
Computer Science Department
King Saud University
Riyadh, Saudi Arabia

Mznah Abdullah Al-Rodhaan
Computer Science Department
King Saud University
Riyadh, Saudi Arabia

Abstract—Diabetes is a chronic disease that needs to regularly be monitored to keep the blood sugar levels within normal ranges. This monitoring depends on the diabetic treatment plan that is periodically reviewed by the endocrinologist. The frequent visit to the main hospital seems to be tiring and time consuming for both endocrinologist and diabetes patients. The patient may have to travel to the main city, paying a ticket and reserving a place to stay. Those expenses can be reduced by remotely monitoring the diabetes patients with the help of mobile devices. In this paper, we introduce our implementation of an integrated monitoring tool for the diabetes patients.

The designed system provides a daily monitoring and monthly services. The daily monitoring includes recording the result of daily analysis and activates to be transmitted from a patient's mobile device to a central database. The monthly services require the patient to visit a nearby care center in the patient home town to do the medical examination and checkups. The result of this visit entered into the system and then synchronized with the central database. Finally, the endocrinologist can remotely monitor the patient record and adjust the treatment plan and the insulin doses if need.

Keywords—Diabetes; Electronic Monitoring; Remote Monitoring, Ubiquities Healthcare; T1DM, T2DM; Android Monitoring System.

I. INTRODUCTION

Diabetes Mellitus (DM) is a chronic disease that is considered to be a metabolic disorder [1]. Diabetes is caused by either the absent of insulin or inability to utilize the produced insulin. Age, family history, body weight, and having previous gestational diabetes are some factors which make the person exhibition for diabetes. Diabetes classified within three categories: Type1 diabetes mellitus (T1DM); which is the result of not producing insulin, Type2 diabetes mellitus (T2DM); which is the result of ignoring insulin by the cells, and gestational diabetes which diagnosed in some pregnancies.

In fact, diabetes needs to be treated either by just modifying life style, or by medications and injections. This treatment is very important to prevent fatal complications. Also, patients who have diabetes need to measure their blood level daily using a glucose meter. In addition, they have to do the HA1c test every three to six months to guide the endocrinologist when evaluating the treatment plan.

Mobile computing can improve the quality of the patients' life by providing systems that help diabetes patients to monitor and control their diseases [2]. These applications can help the endocrinologist by providing a remote monitoring to the patient. Database technologies can be combined with communication technologies to present an integrated diabetes monitoring tool.

Nowadays, many researches are proposing different architectures and system designs for diabetes monitoring system. One of these proposed systems is SMARTDIAB [1] which is proposed by S.G. Mougiakakou et al. in 2010. They combine state of the art approaches in database technologies, communications, simulation algorithm and data mining. SMARTDIAB was consisting of two units: patient unit and patient management unit. The patient unit feed the system by passing information and gets a response regarding to that information.

In this paper, we will improve the SMARTDIAB by building a system of three units: patient unit, endocrinologist unit, and general physician unit. In addition, we increase the strength of our proposed system by concerning the HA1c test's result which helps the endocrinologist to efficiently evaluate the treatment plan. The accuracy of the data is achieved by depending on electronic data input for blood glucose levels' readings. And since the authority is a sensitive criterion in any health care monitoring system, we define three actors in the system with three associated access rights.

This paper is organized as the following: Section 2 discusses the existing work and shows the literature review. Then section 3 presents the problem statement and discusses the proposed system architecture. In Section 4, Patient side solution is presented. After that, in Section 5, we present the doctor side solution followed by discussion about the central database in section 6. In section 7, we show the system evaluation. And then, we show the system strength in section 8. Finally, we summarize our work and highlight some future works in section 9.

II. LITERATURE REVIEW

Diabetes is known as one of the most common diseases that has significant burden on patients and healthcare systems. Nowadays, there are lots of researches in the field of diabetes monitoring. These researches are coming as a sequence of evolutions. The first evolution in diabetes monitoring was the use the computers to manage patient data and save their

records which include personal information, treatment progress and historical information. Then, these monitoring systems were developed and become as dual sides systems. In this type of systems, diabetes can be controlled remotely by which called tele-monitoring and tele-medication. In such systems, the health care providers offer tele-support and monitoring services to patients through a desktop computer at home. Patient also can enter data about his daily intake food, activities, and medication and get a right advice about his condition. Later, with the huge growth in technology, a tele-monitoring system built for mobile phones and PDAs have appeared. These systems can support the patient while he is in move, and supply him with interactive conversation with the healthcare providers. This last type of evolution is now taking a considerable place in researches and markets. And if they well designed, they can provide a really prevention or improvement of diabetes.

In this section, we study the literature and application of diabetes monitoring systems with more details in the last evolution that is related to our work.

A. One-Side Desktop Diabetes Monitoring System

A system was developed by S. Pruna et al. to enable the monitoring of clinical care in the countries of the black sea[3]. The system improves the quality of diabetes services by providing the clinicians with a computerized diabetes registry. The clinicians have many options for the management of the creation, correction, and visualization of patients' records. The system was developed using a modular design and object oriented method approach and its architecture was based on the Good European Health Care Record (GEHR). They use a MS access related tables to store information in the database. Black Sea Tele Diab has two security levels: function level, which assigns specific functions to the user depending on his access right, and data level which restrict the access to the items depending on the user rights.

B. Dual-Side Desktop Diabetes Monitoring System

H. kwon et al. designed new diabetes monitoring system model online using the internet[4]. By using this system patients can contact physicians online, to provide them with information and receive recommendations. In this study, a randomized clinical trial involving 110 patients were conducted to do a comparison between patients who used the internet based blood glucose monitoring system and usual out patient who received the traditional out patient management system for the same period. They call these two groups intervention and control group respectively. Patients at intervention group could access the system via the internet and sent information about blood glucose level, drug information, medication intake, blood pressure and weight. Also, they could ask questions and post comments (e.g. diet exercise, hypoglycemic event). On the other hand, patients were able to see recommendations as well as laboratory data.

C. Mobile and PDA Diabetes Monitoring System

A study and a pilot testing of mobile based remote patient monitoring system was developed to improve blood-pressure (BP) control of hypertensive patient with diabetes [5]. The system was developed within two phases. The first one was to

design a home BP tele-management system. This involved a series of focus group meetings with patients and care providers to guide the development of the system. In the second phase, a pilot study of the system was done. Thirty three patients with type2 diabetes and uncontrolled ambulatory BP were chosen to be under the pilot test. The system was architected into the patient component, a data center and decision support system, and the care provider for physician for reporting and alerting components. Data from the BP monitor device were transmitted via Bluetooth to a programmed mobile phone. The phone securely transmits this data to sever to be stored in a central database. The system applies a set of clinical rules on the data then sends secure written progress messages automatically to the mobile phone.

A. Kollmann et al. [6] find that mobile phone can be used to provide a ubiquitous, easy-to-use, and cost efficient solution for management of diabetes mellitus type1 (T1DM). They do a feasibility study to see how much mobile phone-based data service could be accepted with diabetes mellitus type1 patients, and how much the services can assist T1DM patient on intensive insulin treatment. For this study, researchers had developed software called Diab-memory to support patients entering their information such as: blood-glucose level, injected insulin doses, food intake, well-being and physical activities. Then, data were remotely synchronized to a central database. The system was based on Java2 Mobile edition (J2ME) and built using state of the art internet technology. The study sample was 10 patients with T1DM. Mean age was 36.6 years (± 11.0 years) being in the trail study for three months. The result was focused on patients' adherence to the therapy, availability of the monitoring system and the effects on metabolic status. As questionnaire shows, the system was accepted in general.

Another study was focusing on the difficulty of managing diabetes [6]. In this study, an appropriate diabetic treatment is needed to manage the diabetic's blood sugar level and prevent the future complications. Diabetes patients can send their information such as blood sugar level; blood pressure, food consumption, exercise and then the system manage the treatment by recommending food intake, physical activities, insulin dosage...etc. In order to have the best treatment recommendation, the system is based on rules and the k-Nearest Neighbor (KNN) classifier algorithm. Rule based interface select the knowledge similar to the human experts and it needs pervious knowledge to make rules. The KNN is a machine learning algorithm used in the system to classify results using given data by evaluate time, blood sugar, blood pressure, number of meals, amount of exercise and target caloric consumption. The system integrates the KKN and rule-based inference to generate decisions outside the strict rules. Also, a blood sugar monitoring system for diabetes patient is implemented using web services and Personal Digital Assistant (PDA) programmed in java.

A platform was designed to support the monitoring, management and treatment of Type1 Diabetes Mellitus (T1DM) patients. They call this platform "SMART DIAB" [1]. In order to provide the intelligent monitoring and management, "SMART DIAB" combine many approaches in the field of database and data mining technologies for

management of diabetes data, simulation algorithm for insulin treatment optimization, and communication technologies to implement a tele medicine platform either with wired or wireless capabilities. By combining the previous technologies, SMART DIAB allows intensive monitoring of glucose levels, diabetes treatment optimization, continuous medical care, and improvement of quality of life individuals with T1DM.

D.L. Katz and B. Nordwall state that self-care is efficient to controlling in chronic disease through patient empowerment and timely feedback [7]. They have built wireless system which provides a remote patient monitoring. The system was based on mobile phone technology. Patients are allowed to submit a new data daily and upon these data the server will generate and send feedback messages to the patients' cell phones. The data accuracy ensured by reducing manual data input by integrating the phone with the glucose meter, blood pressure and weight measurement. This platform has some disadvantages as mentioned in [8] such as the limited scope to manually enter data about physical activities and intake food.

Pilot controlled trail were selected to evaluate the feasibility and efficiency of the system to manage type2 diabetes. The selected patients were divided into two groups; 15 patients were using the cell phone technology and 15 patients were in a control group. The study lasts for three months and the researchers collected some improvement feedbacks from the patient in order to apply these modifications to the system. All necessary improvements were done to the system.

III. PROBLEM STATEMENT AND PROPOSED SCHEMA

In this section we present the problem statement and the proposed scheme. First, we define the problem and present the real impact of diabetes on the human beings. We emphasize on the effort that rise from the regular medical examinations and tests. Next, we present our proposed scheme in details.

A. Problem Definition

Since diabetes mellitus is a chronic disease, it needs a regular monitoring to control blood sugar level. Seriously, this monitoring is very critical to prevent many fatal complications. According to the World Health Organization (WHO) [9], there are more than 220 million people worldwide having diabetes. In 2004, about 3.4 million people died from complications and consequences of high blood sugar. WHO expected that diabetes deaths will double between 2005-2030. For the economic side, diabetes mellitus economically has an impact on individuals, families, health systems, and countries.

Diabetes mellitus monitoring includes maintain healthy weight, physical activates, and regular medical examinations and tests. In fact, regular medical examinations and tests seem to be time consuming for both doctors and patients. Patients who live in neighboring villages face some difficulties to go down to the hospital in the city. They have to reserve a ticket or pass a road, reserve a place to stay in addition to the cost of the treatment if any.

Our proposed framework is to produce a diabetes monitoring system based on mobile platform. This system can assist diabetes mellitus patients to monitor their glucose level

according to the glucose level readings, intake food, medication, and physical activities. Our system based on electronic data input, as well as manual data input. This is for accomplish data accuracy and follow-up daily activities at the same time. In addition, clinician-to-patient interaction is allowed by exchanging messages.

The greatest achievement in our proposed system is Arabization. Most of the diabetes monitoring systems was available in English language only, where D.L. Katz and B. Nordwall were translated their system in Spanish [7]. Also we will allow an effective functionality to control the change in HbA1c Levels. Black sea tele Diab [3] was caring about this point, but it was a desktop based system specific for physicians.

B. Proposed System Architecture

The specification of our project is to deliver a smart medical system for diabetes monitoring based on mobile platform. The proposed architecture is consisting of three main units: *the Patient Unit, the Center care Unit and the Hospital Unit*. The *Patient Unit* is the starting point in the system. It contains the patient, the glucose meter, and the mobile device. The glucose level readings are electronically transmitted to the mobile device. Also, the patient can manually enter daily food intake, physical activities and information about some medications and injection. The system may alert the patient about injection time, special medicine reminder, date for medical examination and tests. The patient is able to do a conversation with his/her endocrinologist in emergency cases or for inquiries. These conversations are done via exchanging messages.

The *Care Center Unit* represents the medical center in the patient hometown where the patient can visit instead of the hospital in the main city to do regular checks and medical tests and examinations. The general physician in the medical center is allowed to enter results of the HA1c test.

The *Hospital Unit* is controlled by the endocrinologist who can remotely monitor the patients' status, send advises, and take an action in urgent cases. The endocrinologist is the only actor who allowed modifying the insulin doses regarding to the daily glucose reading and the result of the HA1c test. These three units are integrated together forming a whole Diabetes Monitoring System. The interacting between these units is done by exchanging data. These three units are integrated together forming a whole Diabetes Monitoring System. The interacting between these units is done by exchanging data.

Our proposed system is consisting of three layers: *Presentation Layer, Service Layer, and Data Repository Layer* as shown in Figure 1. The *Presentation Layer* presents the system units' interface. The system user can access the system interface by logging to the system and assigned an access rights. The patient unit is provided by a patient's interface which can be accessed using a mobile device. The glucose readings are transmitted via Bluetooth adaptor to this interface. This patient's interface is presented to the patient by using a micro browser. The endocrinologist and the general physician are provided with a desktop interface. The interface

for these two system units will be presented by using an internet browser.

The *Service Layer* is the core of our system. It performs the services to the system units. It contains two main components: Report Extractor and Smart Analyzer. The Report Extractor service is inquired by the endocrinologist or the general physician. It accesses the Resource Layer and collect information from the requested database. Then, it formats this information and presents it to the requestor. The Smart Analyzer analyzes incoming data before sending it to the database. This Smart Analyzer is depending on some values such as: MIN_Glucose_LEVEL, MAX_Glucose_Level, and MAX_A1c. The analysis in this component is work as the following:

- Once the glucose readings are transmitted from the glucose meter to the mobile phone, they will be compared with MIN_Glucose_LEVEL and MAX_Glucose_Level values. Then, the patient will be advised by the system to do something like: eat more, have a walk, take insulin injection and so on.

When the general physician enters the test result of HA1c test, it will be compared with A1c critical value. If it is above the range, or below it, the endocrinologist will be alerted to make a decision and modify the insulin doses or the treatment plan if needed. His information and presents it to the requestor. The Smart Analyzer is responsible to analyze incoming data before send it to the database. This Smart Analyzer is depending on some values such as: MIN_Glucose_LEVEL, MAX_Glucose_Level, and HA1c value. Resource Layer contains the databases for the system. The system is connected on many databases such as: medical information's database, patients' database, and users' database.

Finally is the *Resource Layer*. It contains the databases for the system. The system is depends on four databases: knowledge database, medical information's database, patients' database, and users' database. The Knowledge database is used by the Smart Analyzer service. It will contain the information needed by this service to perform the analyses such as: MAX_Glucose_Level, MIN_Glucose_LEVEL and A1c. The Smart Analyzer uses this information to compare incoming reading and alert the endocrinologist about up normal or subnormal values. The patients' database holds the patients' personal information such as: patient name, Birth of date, address, telephone number and so on. The Medical Information database is designed to hold medical information for each patient. This medical information include: daily glucose levels, intake food and amount of activities in calories, insulin doses, medical checkups and examinations' results, and information about glucose injection. Finally, the users' database is needed for handling system users' accounts and access rights. The database server is interacting with the web application server.

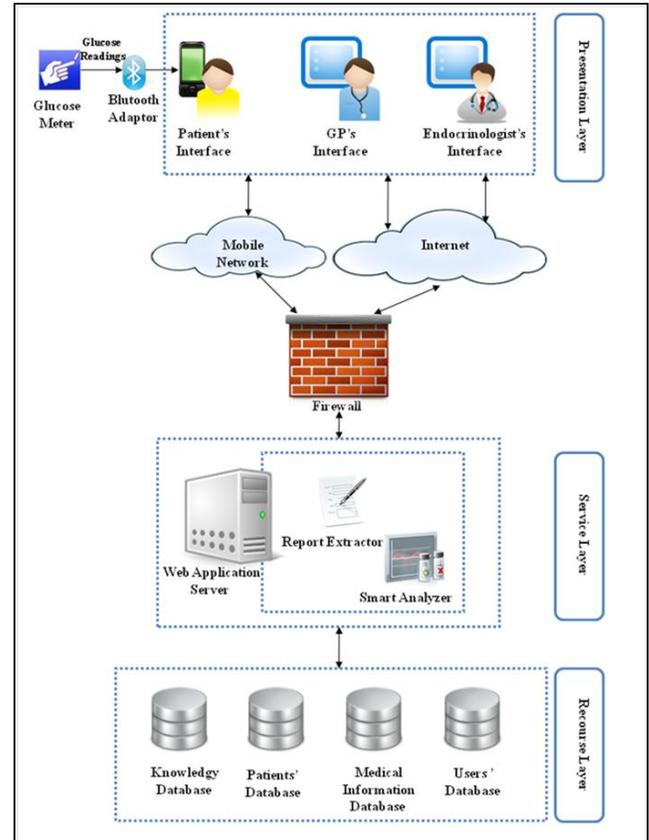


Fig. 1. Proposed System Architecture

IV. PATIENT SIDE IMPLEMENTATION

The Patient –Side Environment was implemented on the android using java programming language. Android SDK was used to setup the development tool and SDK add-ons. There are many reasons behind choosing android OS for the patient-side in our integrated system. We needed to develop a mobile application that can easily communicate with a central database without slowing down the application performance. Also, we needed the mobile to have ability that makes the patient implicitly sends SMS to another phone number and communicates with an external device via a Bluetooth adapter. Moreover, Android supports many libraries that contribute to the implementation process with our integrated environment such as, org.apache.http, org.json, android.bluetooth, android.telephony.SmsManager and java.util.Locale. In the following subsections we will overview some of the used android and java technologies and describe them in more details.

A. Patient side technologies

The first technology is the use of the virtual device configuration. We chose HTC Desire as a target platform and we set the virtual device configuration as SDK Platform Android 2.2, API8. After that, we were able to use the emulator as a virtual device to run our application. Second technology is Dual Language Support.

Our project implementation was designed to support two languages; English language and Arabic Language. Each Patient can set their preferred language. This preferred selection is saved in the database. Once the patient logged in to the system, the preferred language will be retrieved, and according to the preferred language the application will be launched with that language. The design of our system is flexible so it can support more languages in the future. In the database there may be two values for the language: "en" for English language support and "ar" for Arabic language support. After the patient logged in to the system, the Locale will set and configured with the language value. At any time the patient can go to the setting and change the preferred language. This will be updated in the database and the system will be launched with the updated preferred language immediately. We built the interfaces labels and user prompt messages as an XML dictionary for each language. In this dictionary we declare the system interfaces variables and assign the value of this variable according to the target language. For example:

```
<string name="mysevice">My Services</string>
```

"myservices" is a variable for the Button led to patient services screen. Here we set the value as (My Services) for the English language support dictionary and (خدماتي) for the Arabic language support dictionary. This is done for all application interfaces' variables. This led to two dictionaries: en-value and ar-value with more than 27 variables for each.

Bluetooth Connectivity is another technology has been used in our system. The Android platform support exchanging data over a Bluetooth network stack. The Android application access the Bluetooth functionality through the Android Bluetooth APIs. These APIs allow point-to-point feature by performing the following: Query local Bluetooth adaptor, scan for neighbor Bluetooth devices, establish Radio Frequency channels between the local adapter and the selected device, and transfer data to and from other device [10]. Our selected glucose meter has an embedded Bluetooth sensor. It connects to the android application over a Synchronous Serial Port (SSP) profile with a well known Universally Unique Identifier (UUID). The SSP profile is a serial interface used for communication between two devices [11]. The android application communicates with the glucose meter by creating a Radio Frequency Communication (RFCOMM) socket using a well known UUID. And according to the UUID specification for the serial port, the used UUID is 00001101-0000-1000-8000-00805F9B34FB [12]. After RFCOMM socket is connected, the glucose meter is set in pair mode and the android application starts to get the glucose reading by exchanging commands with glucose meter. The outgoing command is written in a created output stream form the connected RFCOMM socket and the responses is read from an input stream created from the same RFCOMM socket.

Also, our system use remote communication with central database. The android application can handle communication with a remote MySQL database to authenticate patient log in information and also to retrieve/ update patient medical information in the database. The communication process with MySQL database is done by posting data using HttpRequest.

First, the application has to make a connection with a PHP script which located in the server. This is done by use HTTP protocol from the android application. PHP scripts are in the middle between the android application and MySQL database. There are many PHP files, each associated with a specific task in the android application such as: authenticate, inter glucose reading, change language...etc. These PHP files are used to call the remote server and POST/GET data to/from it. On the other hand, JSON object is used to handle incoming data from the MySQL database. JSON is Java Script Object Notation used as a data interchange language [13]. Once the query is executed in the PHP file, the result is encoded in string as a JSON representation format. Later, in the Android script, the returned value is handled in JSON array to be decoded and to extract the values from it. These values are used later in the Android application.

Finally, our system can send SMS. The patient can send SMS message to the endocrinologist through the Android application. This can be done by using SMS Manager that manages sending a text message. In the Android script, the endocrinologist's phone number will be retrieved from the MySQL database. Then, the SMS Manger will be called to get the default instance. After that, the function sendTextMessage will be called to send the SMS message.

B. Android Application Activities

In this section, we will preview the features of our implemented Android Application. Android deals with the project interfaces as Activities. The activity is an application component that provides view which users can interact with the application [14].

In our android application there are many activities as shown in Figure 2. The main Activity is named DiabMonSys.java. This activity is responsible to get the user log in information and authenticate this information. After successfully logged in, the patient will be allowed to choose from four features: My Services, My Information, Call Expert and Setting. Each of these features contains sub-functions in deep. My Services can run: GlucoseReadingActivity.java, IntakeFoodActivity.java, PhysicalActivitiesActivity.java and GlucoseInjectionActivity.java.

The same is applied for My Information feature. It can run: MyDosesActivity.java, MyAppointment.java and GlucoseRanges.java. The Call Expert feature is an Activity in its own. We prefer to present it in the main screen because the patient may need it in urgent cases, so it has to be directly reachable.

Finally for the Setting feature, it contains two activities: ChangePasswordActivity.java and ChangeLanguageActivity.java. In the following sub-section we describe each activity in more details.

The *Main Activity* in our project launches log in screen with two data fields, user name and password. When the patient write his/her name in the fields, the Application will connect to the central database, authenticate the input information, and then if the patient authenticated, the patient id with his/her preferred language will be retrieved from the database. The patient id will be used in the next activities, and

the preferred language will be used to set the interfaces language.

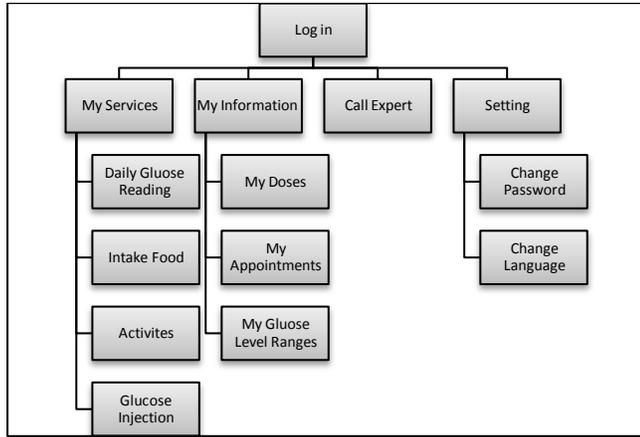


Fig. 2. Patient-Side Android Application's Activities

In *My Services Activity* the patient can select from four sub-activities. First is 'Daily Glucose Reading Activity' which allows the patient to enter his/her daily readings. These readings can be entered either manually or via Bluetooth. After entering the glucose reading, the patient specifies whether this reading is pre-meal or post-meal. This is done because the application will analyze the glucose reading with the time situation and then advice the patient what to do. The analysis of glucose readings is based on standards of medical care in diabetes [15]-[19].

Another service here is 'Intake Food and Physical Activities'. These two activities are based on the patient entry. The patient can select from a menu what he/she was eaten and the amount. The calculation of the calories is based on Food and Nutrition Information Center [20]. After calculation is done, the patient will be advised about what to do e.g.: eat some fruit, drink something, have a walk and so on.

The final service in 'My Services Activity' is 'Glucose Injection'. As a treatment of hypoglycemia the patient have to take the glucose injection. If the patient takes this injection, the endocrinologist may decide to change the next few insulin doses for some time. So, our application allows the patient to enter this information and the endocrinologist will be alerted about that in order to modify the insulin doses if needed.

In *My Information Activity*, we allow the patient to preview the insulin doses, the appointments, and the average of his glucose readings. 'My Doses Activity' will retrieve the insulin doses from the MySQL database and present them to the patient. When the endocrinologist modifies these doses, the new values for these doses will be updated in the patient side as well. Also, the patient will receive the starting time to take this dose and the duration. 'My Appointment Activity' presents the patient with the next appointment. If the status of this appointment is 'urgent', he/she will be alerted about this appointment. The final activity is 'My Glucose Reading Rang'. This activity will show the patient average reading of the glucose for the last 7/14/21 days. This is to give the patient an overall vision about the glucose level.

In addition, sometimes the patient needs a quick advice from his/her endocrinologist. In our application, we allow the patient to send SMS message to the endocrinologist. In the MySQL, each patient is associated with an endocrinologist who follows his/her diabetes. Once the patient starts *Call Expert Activity*, the mobile number of this endocrinologist will be retrieved from the MySQL. And then, the text written by the patient will be sent by the android application to that number. After that, the endocrinologist advises the patient as needed.

Setting is the last activity in the patient side. The patient will be able either reset his/her account password or change the preferred language. If the patient tries to change the password, he/she will be prompted to enter the previous one. If they matched, he/she will be allowed to enter a new one and this new password will be stored in the MySQL database. Also, the patient can select the preferred language. These changes will be stored in the central database for future logins.

V. DOCTOR-SIDE IMPLEMENTATION

In this section we will discuss the implementation environment on the doctor-side Application. Also, we will introduce the system privileges that are supported by the doctor-side application and the functions available for each privilege. Finally we will discuss how we use Java Database Connector (JDBC) in our system to communicate with the central database.

A. Implementation Environment

This side of the application was implemented using java programming. Since this side is straight forward retrieving form the MySQL database and writing to it, the java programming was a good solution for that. The doctor-side application needs an extra library the JDK library which is default with any java application. The extra library is called `mysql-connector-java-x.x.x-bin.jar`, where `x.x.x` is version number. This library is needed to facilitate communication with MySQL database. By using this java library, we can prepare SQL statements and execute it directly within the code. This eliminates the need of PHP scripts. But instead, we divided the functions that perform the tasks as web services. Each of which is specialized to do something in the database.

B. Application privileges

In the MySQL database, the doctor's information contains privilege field. This field may contain one of the two values: 'ENDO' for the endocrinologist and 'GP' for the general physician. If the doctor has 'ENDO' privilege which is the highest scope of functionality. The endocrinologist can have the following functions:

- Preview patient record: shows record number, patient name, patient ID, Date of birth, and phone number.
- Glucose Injection: present the injection dose and whether the patient took it or not. The Endocrinologist has the ability to edit the glucose injection dose.
- Physical Examination: shows the last examination date, the hA1C result and physical examination result. The

Endocrinologist has an ability to enter new test result for a specific patient.

- **Appointments:** shows the last visit, and the next visit. Sometimes the endocrinologist needs to make an urgent appointment according to some results, hA1C result for example, so the endocrinologist will have authority to make an urgent appointment by specifying the date and the priority for this appointment.
- **Insulin Doses:** presents the doses for the insulin injection, the start date for these doses and for how many months. The endocrinologist has the ability to edit these doses and accept the changes.

And as the main idea of the project was to add a general physician role to the system to balance the load with the endocrinologist, the general physician will have a preview privilege with an authority to insert new test examination. Each test examination result is associated with the doctor ID. So, the endocrinologist will know who did this examination for the patient.

C. Java Database Connector (JDBC)

The JDBC API is a java API that facilitates everyday access relational database [21]. By using JDBC the application can connect to MySQL database, send queries and update statements to the database, and retrieve results from the executed queries. In our application, we deal with the database as a separate class which we call "database". The constructor of this class receives the database host as a string variable. Then, a connection with the database will be established in order to be ready for the coming services. This can be done by calling the drive manager of JDBC as the following:

```
con = DriverManager.getConnection(Host, "root", "");
```

Where 'con' is variable connection, 'Host' is the transmitted parameter to the constructor. After that, whatever the services are there will be a function in this class to perform the task. Each service receives a query as a string and some needed parameters to execute the query or the update.

VI. CENTRAL DATA BASE

In our project, we build central databases between the patient-side application and the doctor-side application. These central databases are installed using Wamp Server to synchronise the information between the clients. We have two databases; 'accounts' database and 'dms' database. In our project, there are two databases: 'accounts' database and 'dms' database. 'Accounts' database contains the users log in information for the both application sides, e.g. patient-side application and doctor-side application. These data is used to authenticate users log in information. Once the application is authenticate the user, it will connect with the 'dms' database. The database contains all the system relational tables such as: patient, doctor, appointment and so on.

The databases are installed on a Wamp Server, which is a web development environment that allows creating MySQL databases and applications with PHP [22]. We use this server to build a MySQL relational database. In our project, there are

two databases: 'accounts' database and 'dms' database. 'Accounts' database contains the users log in information for the both application sides, e.g. patient-side application and doctor-side application. These data is used to authenticate users log in information. Once the application is authenticate the user, it will connect with the 'dms' database. It contains all the system relational tables such as: patient, doctor, appointment and so on. An early version of our design was published in [23].

VII. SYSTEM EVALUATION

This section will discuss the evaluation of the integrated system. Our project is considered to be e-health system. And in our evaluation, we will apply the criteria of e-health evaluation [24].

A. Areas of Evaluation

There are some important areas that have to be evaluated to ensure that the system is successful. First is to ensure that the system is useable so the users can achieve the target goal easily. The usability is important in e-health system because if the system is not useable, the patient will never get the benefit of the system. Cost implications are important in evaluation e-health applications [24].

B. Usability Evaluation

Since the target user for our mobile application is diabetes patients, our design should be easy to use and flexible. We put in our consideration that the patient may have vision problems, may be an elderly, or may have an urgent problem. So, in our user interfaces design, we try to follow the mobile application usability check list [25]. Table 1 shows these issues that we considered:

TABLE 1. CONSIDERED ISSUES

No.	What is the Standard?	How to ensue it in our project?
1	Clear and consistent way to go back on every screen	(Back) Button is applied in the activity to go one step back word. (Home) Button is applied in the activity to go the select services home screen.
2	Labels and buttons text are clear and concise	We avoid using much word to explain labels and buttons. We just use the simplest word and well known to the public.
3	Retains overall consistency and behavior with the mobile platform	As android phones are considered to be touchable. We take this consideration in our design and ensure that the touch are working well in our application.
4	Minimalist design - excess features removed	We do not offer more tasks than the minimums the application specification

		needs. This is to help the patient in urgent case that he/she is straight forward.
5	Content is concise and clear	We ensure that advises are understandable and easy to applied.
6	Provides feedback to the user of system status	Our system tells the patient is his/her SMS was send successfully or not. Our system feeds the patient back about his/her glucose insertion in the database.
7	Number of buttons / links is reasonable	There are no buttons that have no significant reason to be.
8	UI elements provide visual feedback when Pressed	The labels prompt the users about what to insert. If the patient put invalid input, he/she will also prompt about this.
9	Colors used provide good contrast	In our design, we put in our consideration the patient cases. We depend on only 4 colors with a blank background. We repeat these colors in same sequence to help the patient self-remembering the task from the color.
10	Colors used provide good readability	We avoid using wooded, striped, nor dotted icons or buttons in our design.
11	Font size and spacing ensures good readability	As we mentioned before, the patient may have vision problems, so we adopt a medium font size that is easy to read.
12	If changes can be made, ensure there is a Save button	In My services activities, the patient can enter new values. In each activity there is a submit button that apply the changes and connect to the database to store the results.
13	Present users with a confirmation option when deleting.	The mobile application side seems to be feeding side, so there are no tasks that require delete operation.
14	Speak the users' language (not technical)	In our design, we use a simple language that is understandable to the public. Also, we present the application in two

		supported language.
15	Error messages are free of technical language	We avoid using technical words to illustrate the errors. Instead we just inform the patient about the error exists

VIII. SYSTEM STRENGTH

Since our integrated system is design as client-server model, it has many strength points that make it better than some existing software. These strength points are:

- **Language independent:** Our system interfaces is design based on language dictionaries. Currently, we support two languages. In future, it can contain many languages just by building the XML language dictionary.
- **Shared Resources:** The database in our application is installed in Wamp Server. The resources in this database are shared between the patient side application as well as doctor side application.
- **Database isolation:** Since the resources in the database are very critical, we limit the insertion to the database according to the authority. System administrator is the one who is able to insert/edit/delete users and equipments information, but not medical ones. The Endocrinologist is the one who is responsible to enter/update the critical medical information, such as: insulin doses, urgent appointments... etc. The GP enters examination results only. This ensures the right person is dealing with the right information.
- **Cost implication:** As Android offers many mobiles with variety in cost. The patient will be able to select a smart phone that is within the budget. Currently, the mobile-side application was tested on HTC Desire which runs on Android 2.2. And regarding the glucose meter, we use myglucohealth glucose meter/ wireless kit, it costs about \$89. This is a mediate cost between the public.

IX. SUMMARY AND CONCLUSION

In this paper, we discuss our implementation of a diabetes monitoring system for managing and monitoring diabetes patients. The specification of our implemented project is based on Android platform. The architecture of the system is depends on three units: the patient unit, the endocrinologist unit, and the general physician unit. These three units are working together forming an integrated diabetes monitoring system.

The Presentation layer contains the actors of the system: the patient, the endocrinologist, and the general physician. The Presentation Layer is followed by the Service Layer: it contains the main two services provided by the system which are: the report extractor service and the smart analyzer service. The report extractor services are responsible for collecting the information and present them to the endocrinologist and the GP along with patients' record and medical history. The smart analyzer service is responsible to analyze the entered glucose

readings and HA1c test result in order to alert the endocrinologist about critical readings. Finally, there is a resource layer which handles the database repository.

Our implemented system provides the patient with many facilities to better monitoring diabetes. Also, the endocrinologist is another aim in our system that we try to reduce the amount of regular medical checkups and examinations done for every diabetes patient. We achieve this goal by adding a general physician who can do this medical checkups and examinations and feed the system with the results. The endocrinologist can access the system and request a report for a certain patient. The endocrinologist can modify the insulin doses and the treatment plan if needed.

Our Implemented system is divided into three environments: Patient-side, doctor-side and central database environments. The patient environment is implemented using Android SDK. In patient-side application, the patient is able to do many services. First, the patient is allowed to feed the system with information through My Services Activity. Patient can enters daily glucose reading, enter intake food, enter physical activities, and enter glucose injection information. Also, patients are allowed to preview their own information through My Information Activity and they can preview the insulin doses, the appointments, and the glucose level ranges. In addition, the patient can make an urgent call to the endocrinologist throw sending SMS message. We add two personal activities which are: change my password, and change my preferred language.

The doctor-side implementation is based on java programming. We use JDBC to facilitate communication with the database. In side Implantation, the interfaces in depending on doctor privilege. The doctor may have 'ENDO' authority which allows them to preview and edit patient record. On the other hand, they may have a 'GP' to preview patient record and enter physical examination results.

In our Implemented project, we use Wamp Server to create MySQL database. Our database is central between the patient-side application and the doctor-side application. We built an administer tool to manage the insert, update or delete users record in the database.

REFERENCES

- [1] S.G. Mougiakakou et al. , SMARTDIAB: a communication and information technology approach for the intelligent monitoring, management and follow-up of type 1 diabetes patients, IEEE Transactions on Information Technology in Biomedicine: A Publication of the IEEE Engineering in Medicine and Biology Society, vol. 14, (May. 2010) , 622-633.
- [2] M. Lee, T.M. Gatton, and K.-K. Lee, A Monitoring and Advisory System for Diabetes Patient Management Using a Rule-Based Method and KNN, Sensors Magazine, vol. 10, (2010) , 3934-3953.
- [3] S. Pruna, N. D. Harris, and R. Dixon, "Black Sea Tele Diab: building an information system for management of diabetes," *Proceedings 2000 IEEE EMBS International Conference on Information Technology Applications in Biomedicine*, Arlington, VA , USA: 2000, pp. 284 - 289.
- [4] C. Cobelli et al. , "Diabetes: Models, Signals, and Control," *Biomedical Engineering, IEEE Reviews*, vol. 2, 2009, pp. 54-96.
- [5] H.-S. Kwon et al. , "Establishment of Blood Glucose Monitoring System Using the Internet," *Diabetes Care*, vol. 27, Feb. 2004, pp. 478 -483.
- [6] A. Kollmann et al. , "Feasibility of a mobile phone-based data service for functional insulin treatment of type 1 diabetes mellitus patients," *Journal of Medical Internet Research*, vol. 9, 2007, p. e36.
- [7] D.L. Katz and B. Nordwall, "Novel interactive cell-phone technology for health enhancement," *Journal of Diabetes Science and Technology*, vol. 2, Jan. 2008, pp. 147-153.
- [8] T. Malasanos, "ANALYSIS: mobile phones integrated into diabetes management: a logical progression," *Journal of Diabetes Science and Technology*, vol. 2, Jan. 2008, pp. 154-155.
- [9] "World Health Organization". Internet: <http://www.who.int/en/> , 2011 [June, 7, 2011].
- [10] Android Developers: Bluetooth. Internet: <http://developers.android.com/guid/topics/wireless/bluetooth.html>, (2012) [January, 4, 2012].
- [11] Microchip Synchronous Serial Port, Microchip technology Inc, (1997), 15-30.
- [12] "Class UUID". Internet: <http://www.avetana-gmbh.de/avetana-gmbh/produkte/doc/javax/bluetooth/UUID.html>, 2012 [January, 4, 2012].
- [13] "Introducing JSON". Internet: <http://www.json.org/>, [January, 4, 2012].
- [14] Android Developers: Activities. Internet: <http://developer.android.com/guide/topics/fundamentals/activities.html>, (4 January 2012) [20, February, 2013].
- [15] Standard of medical care in diabetes, *Diabetes Care*, vol 35,(January 2012), S11, S63.
- [16] "National Diabetes Information Clearinghouse (NDIC)". Internet: <http://diabetes.niddk.nih.gov/>. 7 December 2011 [4 January 2012].
- [17] "Living with Diabetes". Internet: <http://www.diabetes.org/living-with-diabetes/> , 2011 [20, February, 2013].
- [18] A. Rubin, "Diabetes for Dummies, 3rd Edition", Welly Publishing, Inc., 2008.
- [19] "Medicine Net: Hyperglycemia". Internet: <http://www.medicinenet.com/hyperglycemia/page2.htm>, 2012 [20, February, 2013].
- [20] Food and Nutrition Information Center. Internet: http://fnic.nal.usda.gov/nal_display/index.php?info_center=4&tax_level=1 , 27 December 2011 [20, February, 2013].
- [21] "JDBC database Access". Internet: <http://docs.oracle.com/javase/tutorial/jdbc/overview/index.html>, 2011 [20, February, 2013].
- [22] WAMP server. Internet: <http://www.wampserver.com/en/>, 2012 [20, February, 2013].
- [23] Mashael S. Bin-Sabbar and Mznah A. Al-Rodhaan "An Integrated Monitoring System for Managing Diabetes Patients Using Mobile Computing Technology." The World Congress on Engineering and Computer Science (WCECS 2012), San Francisco, USA, October, 2012.
- [24] Q. Le, Evaluation of E-health, Honors thesis, University of Tasmania, (2007).
- [25] Mobile Application Usability Check List, Internet: <http://www.keepitusable.com/keepitusable-mobile-app-usability-checklist.pdf>, (2011) [20, February, 2013].