

A Symbian-based Mobile Solution for Intra-Body Temperature Monitoring

Orlando R. E. Pereira

Department of Informatics
University of Beira Interior,
Covilhã, Portugal
opereira@ubi.pt

João M. L. P. Caldeira

Instituto de Telecomunicações /
University of Beira Interior,
Covilhã, Portugal
EST - IPCB
Castelo Branco, Portugal
jcaldeira@it.ubi.pt

Joel J. P. C. Rodrigues

Instituto de Telecomunicações /
Department of Informatics
University of Beira Interior,
Covilhã, Portugal
joeljr@ieee.org

Abstract— Biofeedback data acquisition is an extremely important task in body sensor networks (BSNs). Data collected by sensors need to be processed in order to be shown in an easy and meaningful way for the user. The use of mobile devices may support and offer new user experiences. When connected to a BSN they can aggregate and process data collected by each sensor, providing a mobile solution for a healthcare system. This mobility offers a better patients' quality of life allowing a regular daily routine and always under monitoring. This paper proposes a Symbian-based mobile solution for intra-body temperature monitoring. Mobile device connects wirelessly to an intra-vaginal temperature sensor and interacts with sensor for temperature data collection and monitoring. This system helps women to detect their fertile and ovulation periods by the increasing of their intra-vaginal temperature. The mobile system was tested and validated with success and it is available for regular use.

Keywords- *Body Sensor Networks; Healthcare Applications; Mobile e-Health; Mobile Monitoring Application*

I. INTRODUCTION

The continuous evolutions in body sensors networks (BSNs) and sensors increase the amount of solutions that can be used to monitor physiological and biological parameters in a single human body [1-4]. To control some disease or human body phenomena, medical staff often uses these parameters [5]. In certain cases, patients (themselves) can make the control of the monitoring parameters at their own homes. Thus, it increases the quality of life of the patients that do not need a continuous medical assistance. The use of suitable body sensors allows the collection of the parameters onboard. Other sensors need an external connection to a base station that saves the acquired data. Monitoring and processing this data by body sensors may be performed either in real-time or in off-line modes. The real-time mode needs the use of a device being always connected to a BSN [6]. The widespread use of mobile devices (with increasing computational power) leads into the enlargement of the frontiers in the application of these platforms for non-standard tasks. Recent projects take advantage of these devices and their features as portable biofeedback monitoring systems [7-9]. The use of mobile

devices became naturally as a great solution for patient's continuous monitoring situations. Therefore, patients only need a regular cell phone to monitor themselves the parameters under control.

Intra-vaginal temperature is one of the most used parameters in women fertility control. It is used to determine the occurrence of the fertile and ovulation periods. In the related literature, it is highly explained the increasing of women core body temperature when this two periods occur [10]. The effective detection of this parameters' increasing could help women to identify their fertile periods. This is highly important in pregnancy issues, once the literature refer this is the period with most probability to get pregnant [11].

This paper presents a mobile solution, based on Symbian operating system (OS), to monitor women's intra-vaginal temperature. The system architecture comprises the following two main modules: *i*) an intra-body sensor for temperature acquisition and storage, and *ii*) a Symbian OS device with the proposed mobile tool for temperature collection and monitoring. The system allows real-time and off-line operating modes, aiming to help females detecting their fertile and ovulation periods in a simple and autonomous way. Furthermore, monitoring the intra-vaginal temperature, it also may be used for preterm labor prevention and detection of pregnancy contractions.

The remainder of this paper is organized as follows. Section II reviews some related works highlighting the problematic of measuring female's temperature for fertility control. Section III describes the system architecture, while Section IV presents the proposed eHealth mobile tool. Section V concentrate on the system validation, focusing on user interface and deployment. Section VI concludes the paper and points out further possible research directions.

II. RELATED WORK

Mobile systems are often used as biofeedback solutions. Several projects were proposed in recent years with this purpose. This section will review some available projects on this topic. It also focuses on some ongoing projects for

monitoring women's intra-vaginal temperature and its problematic.

In [12], a mobile solution for three-tier biofeedback data acquisition and processing is presented. The use of a three-tier architecture allows separation and optimization of system functionalities. It uses body sensor nodes, a sink node for data aggregation from all sensors, and a mobile device responsible for sensors configuration, and processing and visualization of collected data.

In [13], a mobile system for body temperature and electrocardiogram (ECG) monitoring is proposed. This approach uses an SMS-based (short message service) system. It allows the transmission of the patient's data (temperature and ECG) through SMS messages to remote devices anywhere. In [14] the same author proposed a very similar system for body temperature and ECG monitoring. This second proposal differs from the first one on the use of multimedia message service (MMS) to send data to remote devices instead of SMS messages.

Body skin temperature changes with environmental conditions [15]. The use of this parameter to control intra-body phenomena leads into wrong analysis. Female's fertile periods could be detected based on variations of core body temperatures. In [16], the authors addresses issues related with the acquisition of this parameter and proposed a sensor for intra-vaginal temperature acquisition. In our system we use this sensor to collect and monitoring the intra-vaginal temperature measurements. Next section will present the proposed system architecture.

III. SYSTEM ARCHITECTURE

This section describes the mobile system architecture. The proposal is based on a three-tier body sensor network (BSN). It consists on a mobile device, sensors, and the sink node. The sensors collect data from a person and send them to the sink node. The sink node can be observed as a "man in the middle" between sensors and the mobile device, providing constant monitoring of health parameters. It aggregates and storage the collected data from sensors on an internal solid state drive (SSD). The mobile device connects to the sink node via Bluetooth, providing a mobile way for data gathering, processing, and visualization. Figure 1 illustrates a scenario of this architecture.

The proposed architecture allows extraction of system functionalities. Each layer is responsible for an individual task. Sensors are responsible for data collection, sensing health parameters of a given person. The sink assumes the data collection and storage, protecting them from the external world and non-authorized accesses. The mobile device (phone) assumes data processing and presentation (or visualization) of all received data. Consequently, if the mobile device malfunctions, the connection is broken or is not present, but data are not compromised because they will be storage in real time on the sink node. In addition, the mobile device is independent from the involved BSN technologies

because it may use a wired communication, a 2,4GHz wireless radio or any other. The only thing that really matters for a perfect data gathering, processing, and presentation is the type of connection between both mobile device and sink node.

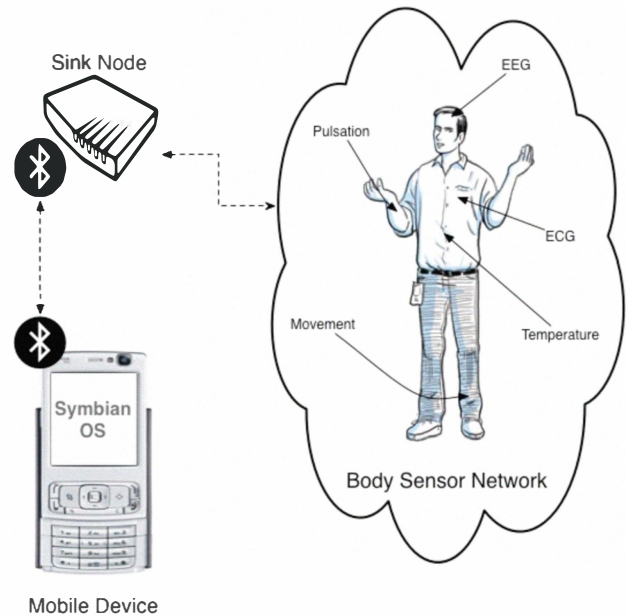


Fig. 1. Illustration of the system architecture.

This proposed system architecture and its development, testing and validation are easier to accomplish because every module can be worked and tuned as a single part. The BSN is protected from the outside world through a specific connection and a given password. This password is a prerequisite for device pairing. It can be applied anywhere even if a mobile device for data processing and presentation is not allowed. The sink node of this prototype is very small, offering the possibility for being placed on small places, such as patients' belt or pocket. Such scenario offers a mobile and portable solution for real time data monitoring, processing, and visualization.

IV. MOBILE DEVICE TOOL

The main contribution of this paper is the proposed mobile system. This tool communicates with the sink node through a Bluetooth connection. The sink and the BSN are protected through a unique password that is hard coded. Using the correct password will give access the Bluetooth connection, data acquisition, analysis, and presentation.

On the development processes our main focus were on system robustness, convenience, and user-friendliness. The data gathered are critical to human life and behavior, therefore very sensitive too, so the application must be robust. The mobile tool must perform robust Bluetooth connection between mobile and the sink node, thus checking for errors on every single stream received with a very tight control over the communication.

The construction of this tool on a mobile device rather than

a portable computer application brings significant advantages, given the popularity of mobile phones and portable digital assistants (PDAs) in the general population. Nevertheless, the tool should be as much simple and easy to use as possible. After initial communication, the tool runs by itself, continuously collecting, processing, and presenting collected data. Furthermore, the same mobile device is not attached to a single patient and it can be applied to gather different patient's data.

Through the use of mobile devices in terms of user friendliness, the display size, the keypad, and battery life are the main limitations presented. The standard vertical display dictates a restriction but it can be changed to display data on horizontal mode and, by this way, we get a guided user interface (GUI) equal to a personal computer. The tool adapts itself automatically to different screen sizes, scaling font size (in six types – title, normal, symbol, dense, legend, default), and graphical scale (based on screen size and text occupation). To achieve the full platform compatibility, no proprietary software development kit (SDK) was utilized. In mobile equipments only numerical and directional keys are used, since those keys are available on all devices.

The application has two main functional blocks, the communication and presentation block. The first block assumes all the Bluetooth communication with the sink node, featuring connection establishment and management. This block has all the necessary configurations to successfully pair both mobile device and sink node. The second block deals with data analysis, processing, and presentation to the user. Every aspect regarding the data is performed here.

To increase the robustness of the mobile tool several exceptions can be thrown in execution time. Each exception will give specific information of the error that has occurred. Since the communication is Bluetooth based, the system wanted to give it the most secure, precise, and robust system as possible. Regarding the communication, exception 1 occurs when the device is not paired with the sink node, because this is a necessary condition to use a Bluetooth connection. Exception 2 appear when the device cannot find or pair with the sink node, while exception 3 happens when the sink node is already in acquisition mode, transmitting data to the mobile device and user tries to start it again. Exception 4 takes place when the mobile tool tries to receive data from the sink node and it is not on acquisition mode. Exception 5 occurs when communication is tried with the closed port while exception 6 is thrown when a communication is in progress and the mobile tool tries to closes it. Exception 7 happens when Bluetooth connection is interrupted, or sink node battery is low/exhausted. Finally, exception 8 is a special case since it has special responsibilities and actions. It occurs when the data received is or may be corrupted. Since all data gathered is scanned before presentation, it turns this exception a passive one. It will not stop the program but in its place, it will inform the user about some potential corrupted data, storing it for analysis and let the program keep running. It is very important

to keep record of all data gathered in order to find some kind of pattern, even if the pattern is from possible corrupted data. Following, in Figure 2 there is a graphical representation of the data flowchart of the mobile tool.

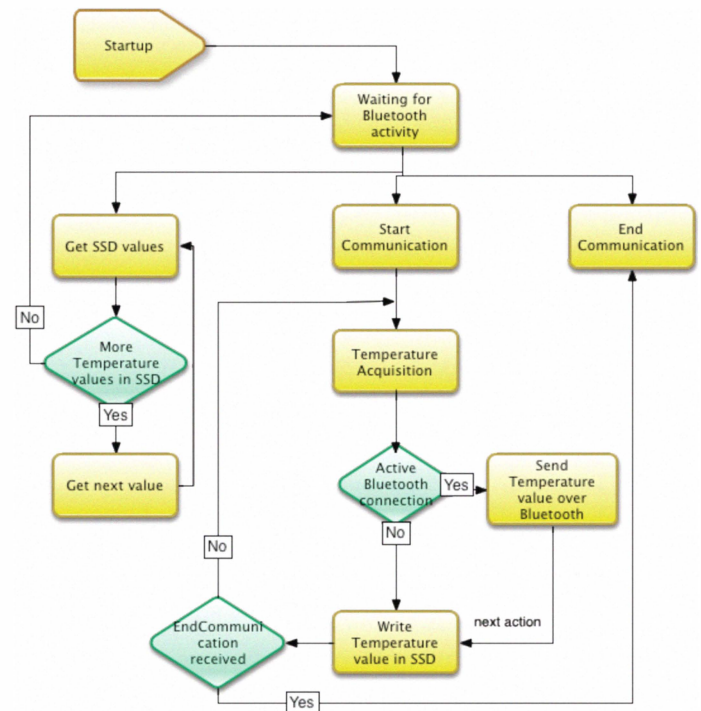


Fig. 2. Mobile tool data flowchart.

The mobile tool starts with a connection to the sink node. After this initial step, the user has two main options to choose. Then, the user can get the data stored on the SSD from previous monitoring or bypass it and start with the real time monitoring. If the user selects the SSD option, he will get all the values saved on it and collected data are graphically displayed if a real time monitoring was disabled. The other option is the collection real time data. When the user chooses this option, a live monitoring will start running. Monitoring functionality can be started on any of compatible device, but only a single sink device (BSN) can be monitored at a given time.

In this approach, data flows in streams. Each stream has a specific data format. By default, the sink node captures and sends the streams with a frequency of 1 second. In terms of data presentation and visualization, one main screen was defined. The screen presents a similar graphical representation to an oscilloscope of the monitored signal.

In terms of programming language, Python was the primary choice for developing this mobile tool instead of pure Java Mobile Edition (J2ME) because of its inherent characteristics. Nokia maintain an interpreter for the Python programming language that is compatible with their 60 series (S60) operating system (OS). The interpreter is signed by Nokia and therefore can be installed on any S60 compatible device. This is the base requirement to allow this tool to run on a mobile

device. It allows the creation of Python applications that do not need to be signed, as they are not directly interacting with the OS. Python applications have more control over the device than applications running in the Java Mobile Edition. For instance, using the S60 Python application-programming interface (API) it is trivial for a script to do a phone call. The biggest penalty for using Python applications on a S60 device is the relative awkwardness involved in launching the Python interpreter and then directing it to the script to be executed. However, the detailed call trace provided by the interpreter in the event of an application fault makes it a very useful platform for development.

Python is a dynamically typed object oriented interpreted language. Applications written in Python may access the functionality of the underlying platform through exposed APIs. For S60 applications there are APIs for graphics, telephony, socket communications, text to speech functionalities, audio recording and playback, etc. Being an interpreted language the data processing performance of Python is less than a native binary application and somewhat less than an application running within a virtual machine; however, the ease of development and detailed crash information make it an ideal choice for prototyping applications.

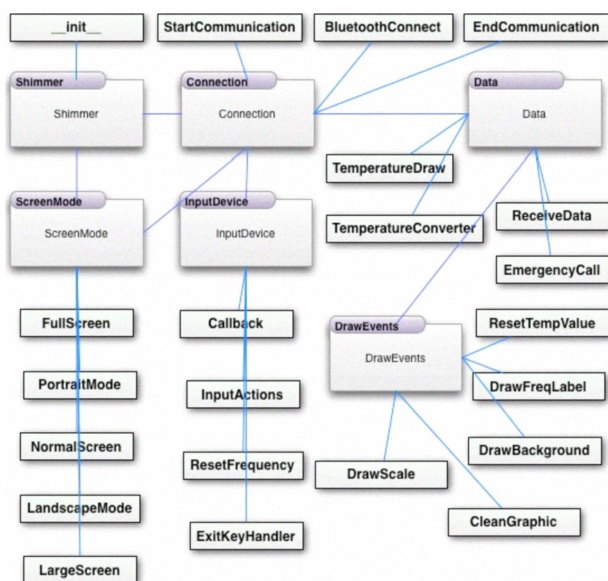


Fig. 3. Mobile tool class diagram (simplified version).

The class diagram of our mobile tool is presented on Figure 3. It is a simplified version, excluding the exception handling classes for the sake of simplicity. Application execution starts at the *Shimmer* class with the default and essential tool configurations, in order to connect to a sink node. After initial configurations (display orientation, application name, fonts settings, screen size detection) the application goes to the *Connection* class. It is responsible for all the pre and post configurations regarding all communications, like Bluetooth address and port configuration.

Data class is the most technology advanced of all because of its inherent characteristics and goals. It is responsible for data analysis, data conversion, and can do an emergency call if some threshold is reached. Our goal was to release an application the most robust as possible so all the data received must be precisely analyzed looking for errors. Every time the tool receives data from the sink it will make some tests in order to verify its authenticity, e. g. comparison of bytes read to bytes that we should read, or order of the received bytes.

DrawEvents class is responsible for converting the received data, in bytes, to graphical presentation. This step has several algorithms linked together to reach the end result that is the correct point to draw on the mobile device monitor. The first step is to strip the correct range of bytes from the stream received to post analysis. This step is essential to security improvement because the received stream has some associated flags. After this, we have guarantees that we are working with the correct data and we can go to the next step. Now, the data conversion to an appropriate temperature value, in Celsius degrees ($^{\circ}\text{C}$), should be performed with another algorithm. All the transmitted data is organized in bytes, and then a transformation is needed. The algorithm must convert a 12-byte value, which corresponds to 4096 single values, in the equivalent temperature value. Afterwards, the temperature is striped and ready for display drawing. This step is very robust and has several related activities because all these activities are done in real time and they are used each time that new data is received. These activities will be reproduced on the display by informing the user of the real temperature, maximum and minimum values or even the data frequency.

The graphical representation of the signal is very important to assist and inform the patient in real time. In order to draw the correct points on the graphic, the calculation of the bottom line of the graphic, based on screen resolution and text occupancy was performed. A transformation function is used to map the 4096 values on a zero to one scale. The normalized value is multiplied by the bottom position of the chart, and a dot represents the current normalized value.

This graphic is totally dynamic and will adjust itself to the situation presented adjusting the Y-axis according to the read temperature. By default, the graphic will have a range of valid temperatures to display but it will automatically adjust to a new scenario if the data received demands. In a given scenario, if the range of temperatures goes from $35,2^{\circ}\text{C}$ to $37,6^{\circ}\text{C}$ and, if the temperature at some point increases to $37,6^{\circ}\text{C}$, the graphic automatically adjusts itself to represent all the data and change the range of temperature to 1°C above the maximum value. The new range will be $35,2^{\circ}\text{C}$ to $38,6^{\circ}\text{C}$. Once again, all these values are totally adjustable, and should be adjustable to a specific case, in order to have a better graphic representation as possible. Another great feature of this class is the ability to apply artificial intelligence and turn it self-sustained, helping a monitored person on a given situation (can be configured according to the case study). On this approach, we are monitoring human body temperature and

if in a given execution point the temperature reaching a high value (configured threshold value), the application will make a phone call to an emergency phone number with a predetermined voice message.

The *ScreenMode* class, as its name suggests, is responsible to handle all screen configurations and layouts, while *InputDevice* class handles the input actions into mobile device display.

The authors of this system believe that this tool presents enough functionality for everyday use. Nevertheless, sometimes, the collected data must be stored for further analysis. In order to support this feature, the sink node includes persistent storage of collected data. The file type, in order to be easily imported to spreadsheet programs, is a text file with coma-separated values (CSV).

V. USER INTERFACE, DEPLOYMENT AND VALIDATION

The user interface was designed to be easy to use and show the needed information in a good meaningful way. Figure 4 presents a screenshot of the instantaneous monitoring signal value. In terms of the mobile tool functionalities, the following two main areas are presented: a text information area and a graphical signal representation.



Fig. 4. Mobile application screenshot of data presentation.

Following the figure, ‘1’ represents the maximum absolute temperature value, while ‘2’ presents the minimum absolute value. Label ‘3’ presents the range values between the maximum and minimum values. Next, ‘4’ depicts the instant temperature value. Number ‘5’ shows the maximum relative value of the temperature that current session has acquired, while number ‘6’ presents the minimum relative temperature value. The number shown in ‘7’ informs about the current data frequency collection. The data can be acquired in several frequency values, ranged from 1 second to 10 minutes (offering the following ranges: 1 second, 30 seconds, 1 min., 5 min., and 10 min.). Finally, the area identified with ‘8’ appears shows the graphical representation of data. When the signal reaches the end of the screen the application does not erase all the signal drawing, but it keeps all the collected data (historic)

for further analysis. Such feature provides a better analysis for a patient.

The mobile tool was deployed, tested, and evaluated on several different devices, such as the Nokia 5320 Express Music (Java-enabled phone, series 60 Symbian OS device version 9.3), the Nokia 5800 Express Music (Java-enabled phone, series 60 Symbian OS device version 9.4), the Nokia 6630 (Java-enabled phone, series 60 Symbian OS device version 8.0), and finally, the Samsung SGH-G810 (Java-enabled phone, series 60 3rd edition, Symbian OS device version 9.2). The application ran effortlessly on every one without any problems.

A big concern during the system construction was the creation of an application that could run on more limited (processor and memory) devices, in order to amplify the range and easy to use of it. On evaluated devices, the less powered was the Nokia 5320 and the application did not step back in comparison with the other mobile devices.

Taking into account that mobile devices are used to make phone calls, send and receive text, and multimedia messages, among other functionalities, the system was designed to cooperate with them without disturb each others. Then, when other functionalities are under use, the system is prepared to run on the background while a person uses the mobile device without problems. On all evaluated devices (above listed) tested with phone calls and SMS’s exchange, as expected, the data continuously flowed from the sink to the application without loosing performance and smoothly.

Table I summarizes the results of a performance comparison between the evaluated devices in five different approaches. The first column has the connection time (in seconds) between the mobile device and the sink. The second column presents the time (in seconds) taken between the initial communication and data presentation. At this time, the tool starts all the initial data processing. These two times were measured based on an average of thirty experiments. The third column informs the smoothness of the application in a range of 1 to 5, being the 1 the worst performance value and 5 the best one. These tests take several parameters in consideration, such as smoothness between menus and graphical interface, temperature signal update, label update, and graphic scaling. The last column presents the multithread functionality. It provides the user capability to make phone calls, text/multimedia messages, listen music or even play games, while the data is continually acquired. In the evaluated equipments, it is available on all of them.

Device	Connection Time (seconds)	Data Presentation Time (seconds)	Smoothness	Multithread
5320	2,1s	1,5s	4,5	yes
5800	2s	1s	5	yes
6630	2s	1s	5	yes
SGH-G810	2,2s	1,5s	4	yes

Table I. Mobile tool screenshot of data presentation (text form).

The mobile application uses approximately 54KB of persistent storage space and 4KB of RAM. It is a small and fast application but a very robust tool. It normally takes around 2 seconds to connect to the sink node and one more second to start gathering, analyzing, and presenting data from sink in a correct way.

Using two mobile devices to communicate with the same sink, the first one that reaches the sink starts the communication with it, while the other is blocked and generates an exception. However, after first device closes communication, other available devices can start communication with the sink. This is a limitation of the current sink implementation that can only serve a mobile tool at a given time, not from the mobile application itself.

In terms of battery power consumption, the application consumes the same when compared to a scenario where a standard Bluetooth connection is used (phone data transfer). However, the constant use of screen backlight can pose significant battery drain (a scenario similar to the use of a screen saver).

Once again, to guarantee that mobile tool performs exactly what is supposed to do, its results were validated against a regular desktop application and, as expected, the results were exactly the same, since the data source is the same. Then, it confirms that mobile application is running properly. The main advantage here is the portability offered by such tool and offers the patient an easy way to view the monitored signals anywhere she wants.

VI. CONCLUSIONS

Mobile healthcare proves big benefits in terms of quality of life for patients who need to be under continuous monitoring for a certain period of time. This paper presented a mobile solution for continuous monitoring of intra-vaginal temperature, using a regular Symbian OS mobile device. The control of temperature helps women in detection of their fertile and ovulation periods. This information represents a key role on fertility issues. A mobile tool was developed to collect, support visualization, and control the temperature in both real-time and off-line modes. The validation of the proposed system using several Symbian OS mobile devices was also performed with success and it is ready for use.

In a near future, the porting of this application to other operating systems such as Google Android OS, iPhone OS, Windows Mobile, and the recent Apple gadget, the iPad, belongs to our next objectives. This extension will allow a broad scope for using this system in more number of available mobile devices.

ACKNOWLEDGMENTS

Part of this work has been supported by the *Instituto de Telecomunicações*, Next Generation Networks and Applications Group (NetGNA), Portugal, in the framework of the Project BodySens, and by the Euro-NF Network of Excellence from the Seventh Framework Programme of EU.

REFERENCES

- [1] K. R. Fowler, "The future of sensors and sensor networks survey results projecting the next 5 years," in *IEEE Sensors Applications Symposium*, (SAS 2009), New Orleans, LA, USA, February 17-19, 2009, pp. 1-6.
- [2] K. Fowler, "Sensor Survey: Part I The Current State of Sensors and Sensor Networks," *IEEE Instrumentation & Measurement Mag.*, 2009, pp. 39-44.
- [3] K. V. Laerhoven, *et al.*, "Medical Healthcare Monitoring with Wearable and Implantable Sensors," in *3rd International Workshop on Ubiquitous Computing for Pervasive Healthcare Applications* (UbiHealth 2004), Nottingham, England, September 7-10, 2004.
- [4] A. Pantelopoulou and N. Bourbakis, "A survey on wearable biosensor systems for health monitoring," in *30th Annual Int. Conference of the IEEE Engineering in Medicine and Biology Society*, (EMBS 2008), Vancouver, BC, Canada, August 20-25, 2008, pp. 4887 - 4890.
- [5] F. Nebeker, "Golden accomplishments in biomedical engineering," in *Engineering in Medicine and Biology Magazine, IEEE*, Vol. 21, May/June 2002, pp. 17-47.
- [6] P. Kulkarni and Y. Öztürk, "Requirements and design spaces of mobile medical care," in *ACM SIGMOBILE Mobile Computing and Communications Review*, Vol. 11, Issue 3, July 2007, pp. 12 - 30.
- [7] T. Gao, C. Pesto, *et al.*, "Wireless Medical Sensor Networks in Emergency Response: Implementation and Pilot Results," in *2008 IEEE Conference on Technologies for Homeland Security*, Waltham, MA, USA, May 12-13, 2008, pp. 187-192.
- [8] D. G. Guo, *et al.*, "A Long-term Wearable Vital Signs Monitoring System using BSN," in *11th EUROMICRO Conference on Digital System Design Architectures, Methods and Tools* (DSD '08), Parma, Italy, September 3-5, 2008, pp. 825-830.
- [9] Y.-t. Zhang, X. Y. Xiang, and C. C. Y. Poon, "The evaluation of nodes of body sensor networks: wearable blood pressure measuring devices," in *Int. Workshop on Wearable and Implantable Body Sensor Networks* (BSN 2006), Cambridge, MA, USA, April 3-5, 2006, pp. 157-161.
- [10] J. Meylan, "Diagnostic Methods in Female Infertility," in http://www.gfmer.ch/Books/Reproductive_health/Diagnostic_methods_female_infertility.html, accessed in January 2010.
- [11] L. Ngalamou and D. Rose, "Fertility Information Appliance," in *15th IEEE Symposium on Computer-Based Medical Systems* (CBMS 2002), Maribor, Slovenia, June 4-7, 2002, pp. 335- 338.
- [12] O. R. E. Pereira, P. A. C. S. Neves, and J. J. P. C. Rodrigues, "Mobile Solution for Three-tier Biofeedback Data Acquisition and Processing," in *2008 IEEE Global Communications Conference* (IEEE GLOBECOM 2008), New Orleans, LA, USA, Nov. 30 - 4 Dec., 2008.
- [13] A. A. Tahat, "Body Temperature and Electrocardiogram Monitoring Using an SMS-Based Telemedicine System," in *4th International Symposium Wireless Pervasive Computing* (ISWPC 2009), Melbourne, Australia, February 11-13, 2009, pp. 1-5.
- [14] A. A. Tahat, "Mobile Personal Electrocardiogram Monitoring System and Transmission using MMS," in *7th International Caribbean Conference on Devices, Circuits and Systems* (ICDCS 2008), Cancun, Mexico, April 28-30, 2008, pp. 1-5.
- [15] I. Campbella, "Body Temperature and its Regulation," in *Anaesthesia & Intensive Care Medicine*. Vol. 9, 2008, pp. 259-263.
- [16] J. J. P. C. Rodrigues, J. Caldeira, and B. Vaidya, "A Novel Intra-body Sensor for Vaginal Temperature Monitoring," in *Sensors Journal, MDPI*, Basel, Switzerland, Vol. 9, 2009, pp. 2797-2808.