Measurement and Visualization of ECG on Mobile Monitoring Stations of Biotelemetric User Adaptive System

Dalibor Janckulik, Leona Motalova, and Ondrej Krejcar

VSB Technical University of Ostrava, Center for Applied Cybernetics,
Department of Measurement and Control, Faculty of Electrical Engineering
and Computer Science, 17. Listopadu 15, 70833 Ostrava Poruba, Czech Republic
Dalibor.Janckulik@hotmail.com, Leona.Motalova@gmail.com,
Ondrej.Krejcar@remoteworld.net

Abstract. Adaptivity and friendliness of user interface is currently discussed in the professional society. Users want a simple, intuitive and graphically attractive interface. On the other hand, it is necessary to change dynamically the user´s experience as they use what best viewing area of the device. The biomedical data are not all the latest knowledge in the area. Our work focuses on exploring the possibilities and the revelation of any deficiencies in the currently used procedures and technologies. The main area of interest of our Biotelemetric User Adaptive System is to provide solution which can be used in different areas of health care and which will be available through PDAs (Personal Digital Assistants), web browsers or desktop clients. In paper we deals with a problem of visualization of measured ECG signal on mobile devices in Real Time as well as with a solution how to solve a problem of unsuccessful data processing on desktop or server. The pre-processing in GSM module processor we choose as a solution of data processing problem.

Keywords: User Adaptivity, Real Time**,** PDA, Embedded Device, Biotelemetry, ECG, Silverlight, WPF, Dynamical Programming, XML.

1 Introduction

The basic idea is to create a system that controls important information about the state of a wheelchair-bound person (monitoring of ECG and pulse in early phases, then other optional values like temperature or oxidation of blood etc.), his situation in time and place (GPS) and an axis tilt of his body or wheelchair (2axis accelerometer). Values are measured with the existing equipment, which communicates with the module for processing via Bluetooth wireless communication technology. Most of the data is processed directly in PDA or Embedded equipment to a form that is acceptable for simple visualization. The next problem is about data processing is ECG packets parsing which can't be processed with real-time response on PDA or embedded device. This restriction can be solved trough the solutions proposed as software in (Section 3) also as hardware solution in (Section 4). The sample architecture of the developed system is sketched on (Fig. 1).

User adaptivity of the system is an important part of different platforms to work with biomedical data. GUI should be realigned according to various parameters and sensed perceptions so that what at first glance most of describing the situation. Our system reacted too to stimuli from the hardware used by devices such as light sensors, accelerometers, GPS and GSM module. The main aim of the platform for patients' bio-parameters monitoring is to offer a solution providing services to help and make full health care more efficient. Physicians and other medical staff will not be forced to make difficult and manual work including unending paperwork, but they will be able to focus on the patients and their problems. All data will be accessible almost anytime anywhere through special applications designated for portable devices web browser or desktop clients and any changes will be made immediately at disposal to medical staff based on the security clearance.

For the physicians is important see the data directly and clearly on the maximum possible viewing area. This problem can be solved by dynamical programming, when we can load only important controls and functional code from database and via dynamicaly controls hiding in GUI on presentation layer. All this possibilities are described in the following sections.
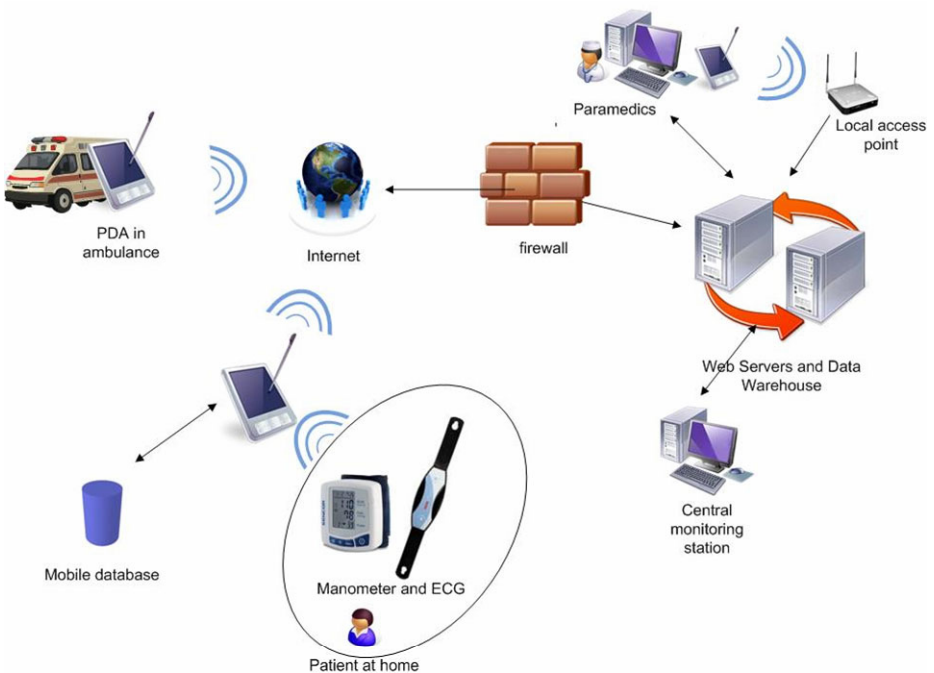
Fig. 1. Sample architecture of one option of system implementation

From the database perspective and bio-signals analysis are the data are stored and automatically analyzed by simply neuronal network.

2   The User Adaptivity of the System

The user adaptivity of the system can be divided into several areas:

- Interaction with the user is based on:
    - used equipment
    - application usage
    - used environment
- Interaction with hardware devices based on the usage of applications

2.1   Adapting Based on the Used Equipment

The application responds as the most of applications on the resolution imaging devices - displays – on the placement of elements. For client applications, where user fills in or views his personal records, the application will behave in different ways while running on personal computer or running on mobile devices such as PDAs with QVGA, VGA or WVGA resolution. For the mobile platform many controls are not available, but we also need to adjust the controls for equipment or touch SmartPhone with a keyboard. Features in one desktop application form must be in the mobile application divided into logical groups. Elderly people may have difficulties with fine resolution and small font, the younger generation will not mind that and we can put in more information into the form.

2.2   Adapting Based on the Usage of Applications in Terms of Interaction with the User

A client application for an average user is responsible for, as already described above, showing data of the user. In the case of desktop, the monitor cannot be easily manipulated and rotated as we like it. In the case of the most wide-spread tablets text information can be shown in more natural form for reading (in height of the book). On the contrary, the graph is better to be displayed in width, so that the longest record gets on the screen. The same goes for using your mobile PDA. This functionality is available for devices with accelerometer, or even for devices without accelerometer, whose display can be rotated by the user's request button.

   One of the novelties is a frequently used battery indicator color change. This is for the battery of mobile devices (tablets, PDA) as well as for the battery of device designed for collecting data (ECG).

   Other features are associated with the popular monitoring of heart rate during exercise. The ECG, which we get we can determine the heart rate. Blue ECG device detects a pulse back. The frequencies of training rate are obtained from the users' data and each zone is indicated by color. So the user can simply by a glance get to know in which zone his heart works.

2.3   Adaptation Based on the Environment

Here the user interface adaptation takes place according to the conditions in which it is used, at night in hospitals we do not want the monitor to light and be disruptive, but

it must be legible. The solution is to switch the colors GUI so that the contrast ratio of application isn't extremely disturbed. The influence is the backlight display device that makes it possible.

2.4   Interaction with Hardware

If the user views only historical records, it does not make any sense to have the Bluetooth on; data transfer is realized using WiFi or GPRS. Compression is related, or transfer of requested data. GPRS is slow, so we transfer only data that the user actually requested, for our WiFi connection bitrates primarily confined. Shutting down the unneeded hardware is today addressed for mobile devices, where device endurance is at one of the first places.

3   Developed Software Parts of Platform

Complete proposition of solution and implementation of the patient's biotelemetry platform oriented for user adaptivity requires determination and teamwork. Every single part of the architecture has to be designed for easy application and connectivity without user extra effort, but user must be able to use given solution easily and effectively. Crucial parts of the whole architecture are network servers, database servers and client applications run-able from standard desktop operating system and client applications for Windows CE based mobile devices.

3.1   Server Part

Database background of solution is built on Microsoft SQL server. One server provides only data warehouse with stored procedures, which represent data interface for other application parts. There are stored all data of medical staff and patients. Data of patients include different records such as diagnosis, treatment progress or data which are results of measuring by small portable devices designated to home care. These data represent the greatest problem, because amount of these data rapidly increase with increasing amount of patients. Due to this fact database servers are very loaded. The stored procedures (programmed by Transact-SQL language) serve basic data parsing on weak mobile devices, which have too much problems with parsing and subsequently visualization of measured data. At this time our team is focused on implementation of analytical and reporting parts for more detailed analysis of measured/collected data. Business intelligence part of SQL server is a powerful tool which allows us to create reports faster than C# application on client side.

The only possibility of running our application on all platforms is an implementation of a view and controller layer as web application. We use two different technologies. ASP.NET is purely for web application (browser independent) and for web services. The second technology is Silverlight (only for Internet Explorer and Mozilla Firefox). Silverlight application is possible to run in Out-of-Browser mode.

In order to run a web server, an operating system supporting IIS (Internet Information Services) is needed. IIS allow to users to connect to the web server by the HTTP protocol. The web service transfers data between the server and PDA/Embedded devices. Web

service also read the data, sends acknowledgments, and stores the data in the database. The service is built upon ASP.NET 2.0 technology. The SOAP protocol is used for the transport of XML data.

Methods that devices communicating with the web service can use include:

- receiving measured data,
- receiving patient data,
- deleting a patient,
- patient data sending.
- RAW data parsing
- other ...

To observe measured data effectively, visualization is needed. A type of graph as used in professional solutions is an ideal solution. To achieve this in a server application, a .NET Chart Control can be used for ASP.NET 3.5. For data analysis, neural nets are a convenient solution. However, there are problems in the automatic detection of critical states. Every person has a specific ECG pattern. The Neural net has to learn to distinguish critical states of each patient separately.

3.2   Desktop Part

The desktop client application is the main and the only part of the entire platform for patients' bio-parameters monitoring, which medical staff uses directly. It is obvious, that if Guardian should optimally replace classic paperwork, simplicity and trouble-free usage of client application are very important factors, which affect whether the doctors and medical staff accept this solution with enthusiasm and solution will be fully used or not. The options of desktop client application have to be easily upgraded. Therefore it is important to reliably design architecture which will allow that. Implementation of user functions is also important. Using the platform.NET in-build characteristics and open standards such as XML, XPath and other, is crucial. Because of that it is easy to configure or upgrade application. User interface is also easy to adjust to user request or clearance. Well designed architecture allows not only easier developing to software engineer, but brings also new and useful functions to the user. The design of appropriate architecture is crucial for the next development of implemented client application, which will be easily upgraded with new functions in the future without making any expensive and demanding changes in programmatic code.

- User Interface - represents the part of application, which is made by components of user interface.
- Command Manager - associates and administrates all existing objects „Command", which execute operations called mainly by user interface.
- UI Factories - dynamic assembling of some parts of user interface during the run of application or immediately after lunch.
- XML Navigators – reading of components of XML documents, which describe user interface.
- Configuration – reading and editing of XML files, which are designated for primary application configuration.

- Web Service Proxy - creates an impression of existence of local web services copies.
- XML - files, which are indispensable for running of entire application. These files control exact syntax, which enables their easy programmatic analysis.

XML represents great role in suggested solution. Options of this technology are used by dynamic assembling key components of graphic user interface, which enables its changing in dependence on roles or clearance of users. It is also used for easy application configuration.

3.3 Mobile Parts

The main part of the system is an Embedded or PDA device. The difference in applications for measurement units is the possibility to visualize the measured data in both Real-time Graph and Historical Trend Graph, which can be omitted on an embedded device. PDA is a much better choice for Personal Healthcare, where the patient is already healthy and needs to review his condition. Embedded devices can be designed for one user, with the option to use an external display used for settings or with the possibility of usage in extreme conditions.

The user adaptivity on mobile devices is control reorganization based on screen rotation provided by operating system. For automatical screen orientation change the mobile device must have accelerometer or G-sensor (HTC feature) implemented. The next step is color theme of application change. This feature is provided by measured level of actual lighting from built-in light-sensor. Devices which do not have this hardware parts can be set to defaults. Application reads default values from registry or from internal mobile database. Dynamically generated design for standard WindowsForm application is developed by dynamical programming technique, where small parts of application or controls source codes are inserted in database. When the application is running, the basic parts are loaded from database. Next choice for dynamically loaded design is XML generated design, described above in (section 3.2.).

Devices based on old PDA type (all devices developed before 2007 too) have a several limitations such as low CPU performance, low battery life or small display, which is possible to solve by embedded version of such mobile clients. We created a special windows mobile based embedded device. During the development process the several problems occurred. One of them and the most important was the need of a new operation system creation for our special architectural and device needs. We used the Microsoft PlatformBuilder for Windows CE 4.2 tools. The created operation system based on standard windows mobile has several drivers which we need to operate with communication devices and measurement devices.

At this time we use modern devices based on Windows Mobile 6 and newer. These devices have no limitations listed above; the only problem is higher price.

Measured data are stored on a SD Memory Card as a database of MS SQL Server 2008 Mobile Edition. The performance of available devices seems insufficient for sequential access (Table 1); parsing of incoming packets is heavily time-consuming. Pseudo paralleling is strongly required. This solution we used only in offline system implementation or when the internet connection is not available. In online (to the internet connected) system we send RAW (unparsed) data to the web service, which

calls stored procedure for bitwise oriented parsing on database server. This "outsourced" data parsing is in a 12-chanel ECG measurement case faster as internal parsing, when is device unusable for overflow reason.

Table 1. Mobile Devices with LCD 480x800 pixels, GSM, WiFi, BT

Mobile device	OS WM	Display	CPU [Mhz]	SPB Benchmark Index
HTC Touch HD	6.1 Pro	LCD 3,8"	528	553
HTC Touch HD2	6.5 Pro	LCD 4,3"	1000	779
HTC Touch Diamond 2	6.5 Pro	LCD 3,2"	528	520
Samsung Omnia II	6.5 Pro	LED 3,7"	800	565

Table 2. BT ECG Device -> Mobile Device measurement

ECG device	Packet Size [Bytes]	Speed [Packets/s]	Transfer Speed [kB/s]
3 Channels	100	3x	0,3
12 Channels	300	5x	1,5

Table 3. BT ECG device measurement real-time packet parsing possibility

ECG device	Platform	Problem	Real Time
BT – Mobile device – Server – Visualization	.NET Framework	Memory overflow	Impossible
BT – Mobile device – Server – Visualization	C++	Memory overflow	Impossible
BT – Mobile device – Server – DB - Visualization	SQL Server procedure	-	Soft RT (2 sec deadline)
BT – MCU - Mobile device – Server - Visualization	MCU HCS08	-	Hard RT

4   Developed Hardware Parts of Platform

One of the main parts of system is a hardware background. If we want to analyze some data, we must measure and process it. This parts are realized over commercial ECG Corbelt, Blue ECG or own built ECG device for measurement and Cinterion AC75 GSM module for analysis (data parsing). GSM module is also used only in the own mobile device solution case.
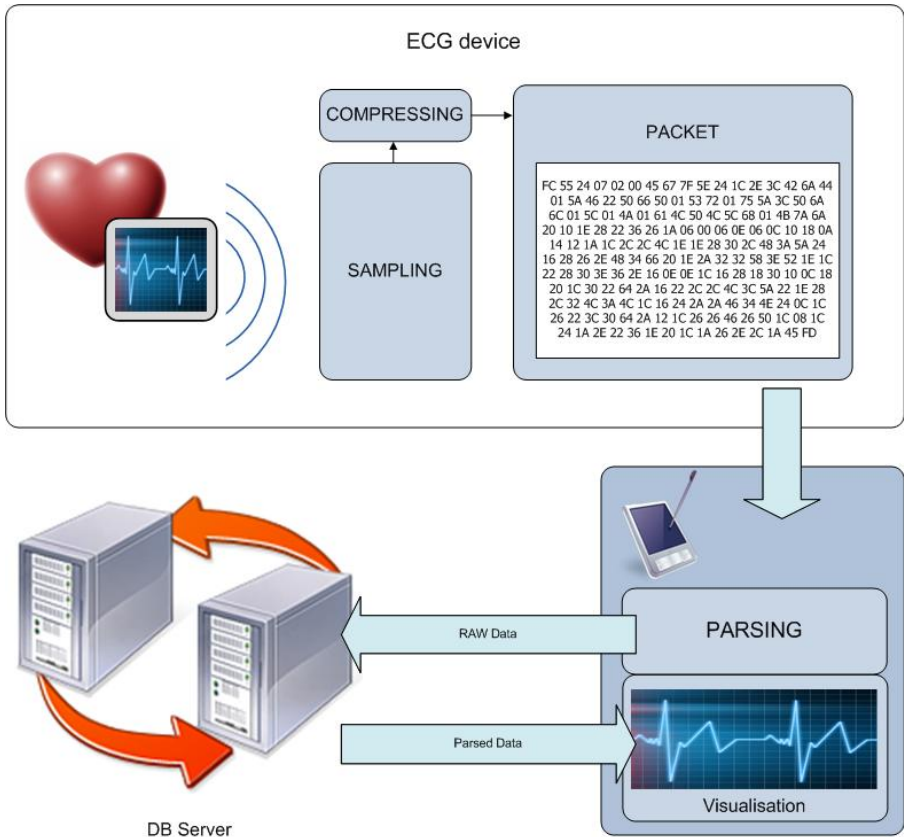
Fig. 2. Measurement, packet creating and packet parsing on mobile device schema

4.1 ECG Device

As measurement device is possible to connect several device with Bluetooth communication possibility. In our application we use an ECG Measurement Unit (bipolar ECG Corbelt or 12 channels BlueECG) through a virtual serial port using wireless Bluetooth technology.

The amount needed to transfer from source device through a Bluetooth is in Table 2. You can compare the increased data transfer speed in case of 12 channels ECG to 1 500 bytes per second. These data amount is very small; on the other hand the data are going as packets, so the processing is needed before the real data can be accessed.

This process (called "parsing") take an unacceptable time in case of mobile device to process the data in Real Time. Same problem is growing on desktop PC, where the C# or C++ is used. In both cases the Memory Overflow is reached. The only possible way we found is in use of SQL procedure which is executed on SQL server. When the data (packets) are stored in table the procedure is call to execute and provide RAW data. In such case the data are ready to user-consumer application until 2 second deadline, so the Soft Real Time is possible to use.

The RAW data table contains full size packets received from an ECG device. Only the packets with measured data are stored to database. Those packets must contain in part of packet number bytes with a value of 0x0724.

The table with parsed data contains decimal values. Column „I" contains data from bipolar ECG; column „II" with „I" contains data from 6-channel ECG. The 12-channel ECG fills after parsing columns „I II V1 V2 V3 V4 V5 V6".

To get a real ECG data immediately after the measurement the next way can be used. We can use a special microcontroller (MCU) embedded in USB unit. This MCU unit has a full speed (12Mbit/s) USB access and BT is connected through a serial port. The MCU unit process all needed operations with parsing to provide a real ECG record to database or directly to visualizing application. In case of WPF application the Hard Real Time mode was reached.

An example of real ECG record is shown in (Table 4) and (Table 5). In these cases, only Soft Real Time mode was reached even when a special MCU unit was used for preprocessing of data. In next subsection the use of WPF application is described as only way when the hard Real Time was reached on windows desktop PC (Windows RTX extension was used).

Table 4. Stored RAW data from ECG device in database table

id	RAW data	usr_id	Stamp	parsed
5238	0xFB3708000...EAC2FDC	13	5.1.2010 12:42:33.45	1
5239	0xFB284A000...0BB7EDC	13	5.1.2010 12:42:33.45	1
5240	0xFB2A3C000...40EA4DC	13	5.1.2010 12:42:33.45	1
5241	0xFB2846000...0DBAEDC	13	5.1.2010 12:42:33.45	1
5242	0xFB2893000...0CB8EDC	13	5.1.2010 12:42:33.45	1
5243	0xFB374B000...4A92FDC	13	5.1.2010 12:42:33.45	1
5244	0xFB2B5D000...80DA2DC	13	5.1.2010 12:42:33.45	1

Table 5. Parsed RAW data stored in database table

id	I	II	V1	V2	V3	V4	V5	V6	original_id
304	9250	7934	8163	5422	5743	8924	7365	8067	10102
305	7932	7396	8534	5709	6108	6385	6313	5893	10102
306	12520	8298	7446	8462	9032	7026	6359	7204	10102
307	7552	6580	6476	7243	9834	8549	7936	6228	10103
308	8976	5783	6948	9684	5638	9336	9316	9746	10103
309	9298	6224	9682	6734	5784	6164	5652	7046	10103
310	9114	9463	7466	4624	8256	9032	8552	8592	10104
311	9773	6350	7581	8779	7365	7231	5137	7295	10104
312	8627	6278	5274	5624	8645	7405	5874	6808	10104

4.2   Battery Consumption Tests

During the real tests the battery consumption tests were executed. Firstly the set of two monocell battery with nominal voltage of 2,5 V were tested without successful time of usage. They provide only 2 hours of operation time. At second case the Lithium-Polymer cell was used with nominal voltage of 3,7 V. In this case an additional circuitry is needed to use an USB port for recharging of battery in device. At figure (Fig. 3) the battery test screen of 12 channels ECG is presented. Figure shows the voltage of 3V (discharged battery) where current is presented by light trace on oscilloscope screen and its average value is approximately equal to 106 mA.

   Figure (Fig. 4) shows the same at a normal charged battery voltage level where the average current is going down to 81 mA. In case of Li-Pol battery usage the operation time of 12 channels ECG is about 10 hours.
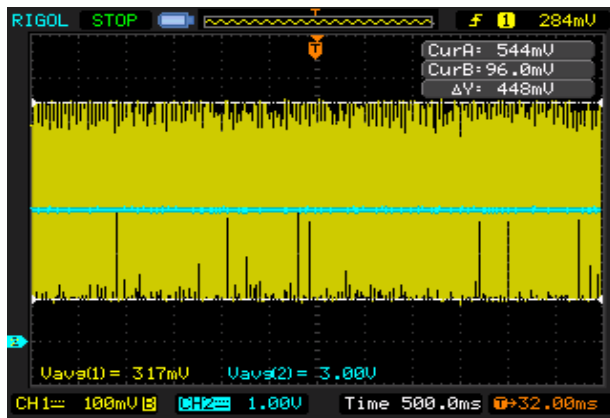

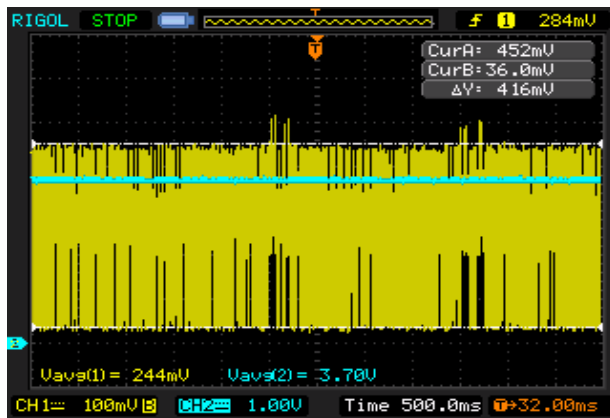**Fig. 3.** Battery test screens of 12 channels ECG. Discharged battery.

Fig. 4. Battery test screens of 12 channels ECG. Charged battery.

4.3   MCU Module and GSM Module

Other solution of parsing problem is a Java application in GSM module. The GSM module cointains own GSM communication module and processor which could be programmed by Java. The consumption of this microprocessor is acceptable, when we equals it to computing performance. Integrated processor have good computing power in bitwise operations, then the visualization application have to more time for own visualization. This is the same solution as special USB Bluetooth dongle with MCU. In other case is possible to the processed data append an informations about GPS position or data from accelerometer to visualize or send position data to user or
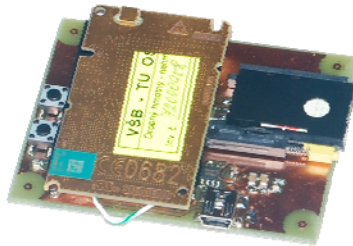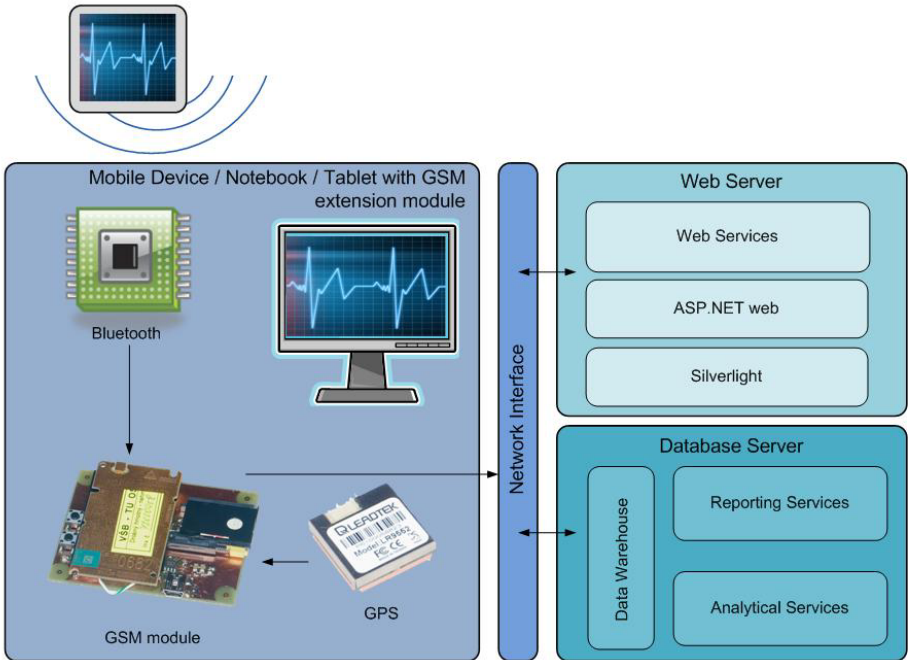

Fig. 5. Purpose-build GSM module

Fig. 6. Communication architecture

patient keeper eg. The special USB Bluetooth dongle consists in the construction of its way "smart" bluetooth module, which, in addition to its own communication interface also included a small single-chip microcomputer (MCU) with no operating system to work according to an algorithm for parsing mentioned packages and working as a parser.

Program in the MCU would work with individual bytes in the packet at the lowest hardware level, which should cause that for these operations will be sufficient very simple, small and inexpensive system with negligible energy demands. Both solutions takes place reliably in real time, and the parser could be equipped with a large enough flash memory cache. This could save the processed data at a time when it is not normal Real-Time operating system was unable to read and save. This would ensure that there is, in short-term 'freeze' the system is no loss of valuable data.

For the soft-real-time solution is here option, send data directly via GSM module to database server and use the stored parsing procedure particularly described in (section 3.1.).

5   User Interface Designing

Proposal for this UI cyclically through several steps:

- Needs Analysis
- Design of UI
- Implementation of UI design and dynamics
- Testing

This is the so-called spiral development model, in which each spiral passage assess emerging requirements identified during testing of the current UI. At the very beginning of the development of UI, we have addressed a number of potential users. Questionnaires were sent to those under which we created the initial list of core requirements for the application and its UI. After the design and implementation of the application was sent back to users (in this case and inexperienced - viz. Testing UI). For testing, we obtained a list of comments on the application of its control and in some cases, suggestions of possible vylepšení.V this case, the tester user because our priority was to create such a UI, which is adapted to the user so that he worked well with him. Currently the application is again in the testing phase (4th course of the cycle) and in the near future we expect the results and any proposals for possible changes.

5.1   Visualization

To make an ECG visualisation the measured data are needed at the beginning. The measurement is made on bipolar ECG corbel and 12 channels BlueECG.

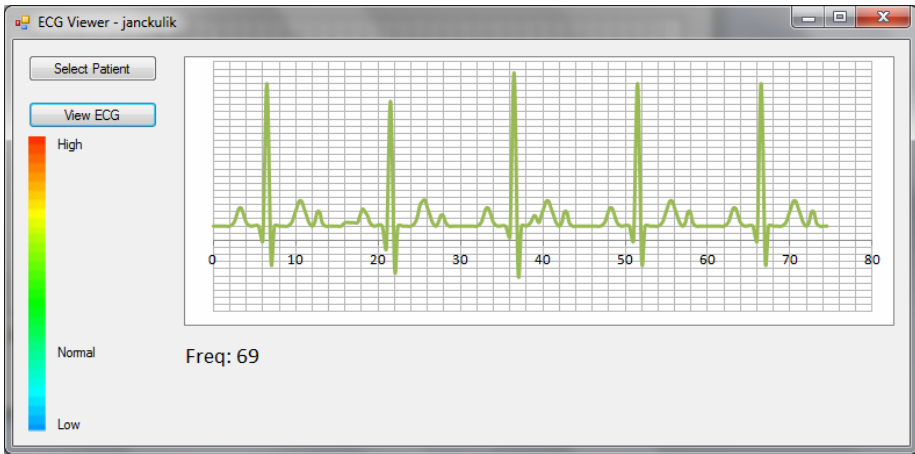**Fig. 7.** WPF ECG visualization application

Fig. 8. UA highlighting via the ECG curve color (normal pulse)

5.3 ECG Visualization in WPF Application

WPF (Windows Presentation Foundation) provide up to date possibilities to visualize ECG data on desktop PC. We create a WPF application to provide a full scale of graphic features to user. WPF technology runs directly on GPU possibilities (Graphic Processing Unit) which is founded on modern graphics cards. This fact is key parameter for speed of data presentation of screen. CPU has more time to compute others tasks (e.g. ECG data analyses by neural network tasks). WPF has a more
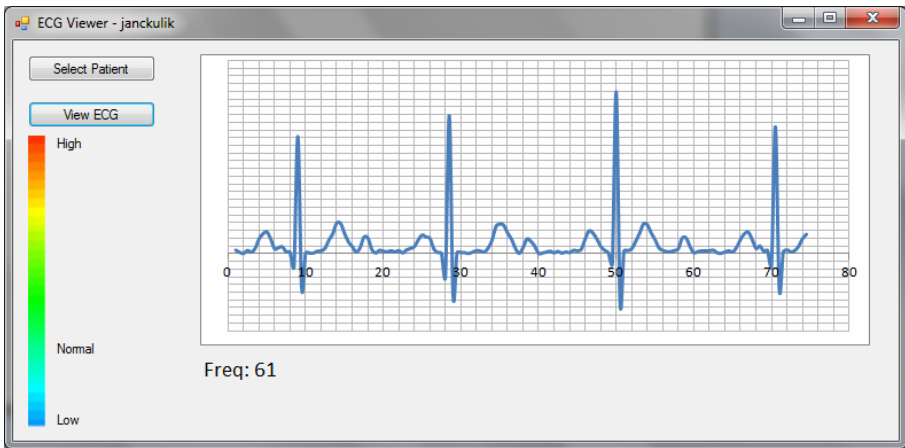
Fig. 9. UA highlighting via the ECG curve color (elevated pulse)

design possibilities in compare to classical Windows Forms including 3D animation,pattern changes of whatever elements etc. WPF application allows viewing an ECG characteristic of measured patient in Real Time, selection of patient from database and view of historical graphs. The figure (Fig. 7) shows an example of bipolar ECG characteristic in WPF application.

6 User Interface Testing

To see our response to user created user interfaces, the application was submitted by fifteen users, divided into three groups according to their degree of user knowledge. Groups are "experienced user", "moderately experienced" and "Inexperienced user". Each user in the UI test assessed three aspects: friendliness, intuitiveness and overall rating. The user-friendliness of judging shows how he liked the graphic design of the program. Intuitiveness describes whether the user explore just guess what to do at that moment. The overall user evaluation of grades describes how he liked the overall application.

The results (Table 6) show that among users who are not accustomed to using similar applications and advanced user, there are only minimal differences. The worst rating of "4" has been achieved in a single user. It was inexperienced user reviews and intuitiveness. Second worst rating of "3" was obtained from a single user type "Inexperienced user", also in the evaluation of intuitiveness. Other users have always evaluated the application of the mark "1" and "2". Overall, all users have responded to the application well, had a problem with its use and its appearance and evaluated as friendly.

Table 6. User Adaptivity classification

Type of User	Assessment of user interface			
	Pleasantness	Intuitiveness	Response time	Summary
Inexperienced 1	1	2	2	1
Inexperienced 2	1	3	2	2
Inexperienced 3	2	1	2	2
Inexperienced 4	1	4	1	3
Inexperienced 5	1	1	1	1
Moderately experienced 1	2	2	2	2
Moderately experienced 2	1	1	1	1
Moderately experienced 3	1	1	1	1
Moderately experienced 4	1	1	2	1
Moderately experienced 5	1	3	1	2
Experienced 1	1	1	1	1
Experienced 2	2	1	1	1
Experienced 3	1	2	1	1
Experienced 4	1	1	1	1
Experienced 5	1	1	1	1

Table 7. Legend to user adaptivity classification

assessment	meaning
1	excellent
2	good
3	may be
4	bad

7   Conclusions

One of the most important parameters in the selection of applications has recently become a consumer society in the user interface. That´s why, in our applications we try to apply as much knowledge and available technology processes for their implementation. The measuring device (bipolar Corbelt ECG and 12 channels

BlueECG) which was tested in extreme conditions in a cryogen room in spa Teplice nad Becvou (Czech Republic) (-136°C), other design features in visualization software on outdoor device must be implemented. For these devices, (with Windows CE operating system) so powerful development tools for easy implementation are not available; as for indoor devices with operating systems clearly specialized for UI design (e.g. iPhone OS). The solution or the simplest way for implementation "handsome" UI for those "oldiest" devices with Windows CE is Microsoft Silverlight technology which is possible startup on Windows CE devices. The visualization of the measured data was reached in case of WPF usage. On the other side, for hardware parts the most important thing is the longest possible battery life for the longest operation time for measurement. Executed battery consumption tests provide a suggestion to use a Li-Pol battery with nominal voltage of 3,7 V. In such case the operation time is going to sufficient 10 hours. As the final improvement in the future, the application would have some special algorithm, which could recognize any symptoms of the QRS curve, and make the job for the doctors much easier.

References

1. Janckulik, D., Krejcar, O., Martinovic, J.: Personal Telemetric System – Guardian. In: Biodevices 2008, Insticc Setubal, Funchal, Portugal, pp. 170–173 (2008)
2. Krejcar, O., Cernohorsky, J., Janckulik, D.: Portable devices in Architecture of Personal Biotelemetric Systems. In: 4th WSEAS International Conference on Cellular and Molecular Biology, Biophysics and Bioengineering, BIO 2008, Puerto De La Cruz, Canary Islands, Spain, December 15-17, pp. 60–64 (2008)
3. Krejcar, O., Cernohorsky, J., Czekaj, P.: Secured Access to RT Database in Biotelemetric System. In: 4th WSEAS Int. Conference on Cellular and Molecular Biology, Biophysics and Bioengineering, BIO 2008, Puerto De La Cruz, Canary Islnds, Spain, December 15-17, pp. 70–73 (2008)
4. Krejcar, O., Cernohorsky, J., Janckulik, D.: Database Architecture for real-time accessing of Personal Biotelemetric Systems. In: 4th WSEAS Int. Conference on Cellular and Molecular Biology, Biophysics and Bioengineering, BIO 2008, Puerto De La Cruz, Canary Islands, Spain, December 15-17, pp. 85–89 (2008)
5. Krejcar, O., Janckulik, D., Motalova, L., Kufel, J.: Mobile Monitoring Stations and Web Visualization of Biotelemetric System - Guardian II. In: Mehmood, R., et al. (eds.) EuropeComm 2009. LNICST, vol. 16, pp. 284–291. Springer, Heidelberg (2009)
6. Krejcar, O., Janckulik, D., Motalova, L.: Complex Biomedical System with Mobile Clients. In: Dössel, O., Schlegel, W.C. (eds.) The World Congress on Medical Physics and Biomedical Engineering, WC 2009, Munich, Germany, September 07-12. IFMBE Proceedings, vol. 25/5, Springer, Heidelberg (2009)
7. Krejcar, O., Janckulik, D., Motalova, L., Frischer, R.: Architecture of Mobile and Desktop Stations for Noninvasive Continuous Blood Pressure Measurement. In: Dössel, O., Schlegel, W.C. (eds.) The World Congress on Medical Physics and Biomedical Engineering, WC 2009, Munich, Germany, September 07-12. IFMBE Proceedings, vol. 25/5. Springer, Heidelberg (2009)

8. Penhaker, M., Cerny, M., Martinak, L., Spisak, J., Valkova, A.: HomeCare - Smart embedded biotelemetry system. In: World Congress on Medical Physics and Biomedical Engineering, Seoul, South Korea, August 27-September 01. PTS 1-6, vol. 14, pp. 711–714 (2006)
9. Brida, P., Duha, J., Krasnovsky, M.: On the accuracy of weighted proximity based localization in wireless sensor networks. In: Personal Wireless Communications. IFIP, vol. 245, pp. 423–432 (2007)
10. Cerny, M., Penhaker, M.: Biotelemetry. In: 14th Nordic-Baltic Conference an Biomedical Engineering and Medical Physics, Riga, Latvia, June 16-20. IFMBE Proceedings, vol. 20, pp. 405–408 (2008)