

Design and Implementation of Obstetrical Central Station Client Based on Android

Huijin Wang*, Chao Han**

*(College of Information Science and Technology, Jinan University, Guang dong,China
Email: twang21cn@21cn.com)

** (College of Information Science and Technology, Jinan University, Guang dong,China
Email: swakyle@126.com)

Abstract:

Multi-bed obstetrical central station system has already been widely applied, while development of Android mobile devices is affecting people's lives, Android platform obstetrical central station clients is important. The client function is realized by defining the communication protocol between client and server. Improve user experience of display by adapting for the screen. Improve performance by using parametric partial drawing. Simplify network exception detection by introducing the heartbeat signal. The implemented client meets the user needs, has good user experience, and helps the company to win a widely attention in a medical exhibition.

Keywords — Android, Fetal monitor, Multi-bed, Obstetrical, Central monitor station

I. INTRODUCTION

Fetal heart rate monitoring has become the world's most common and effective obstetric prenatal fetal monitoring method, it is possible to detect fetal's tolerance to hypoxia, oxygen reserves, fetal distress in uterus, etc., to reduce damage to the fetus caused by lack of environmental oxygen, to improve production quality and reduce perinatal mortality, where the whole production process intrapartum care is more important significance. To implement whole production process monitor, multi-bed obstetric central monitoring network system is the best solution. [1] proposed a multi-bed central station fetal monitoring system, used in many hospitals, the effect is good.

As time progresses, more and more convenient mobile smart devices in line with people's habits is widely used. With respect to the closed Apple devices and the expensive nature of Windows devices, android device gain a larger market for its openness [2].

ECG monitor on Android phone has been implemented. [3] proposed an system that Android device act as a remote ECG monitoring detection

gateway. [4] proposed a fetal monitoring system using NFC. Both these monitoring systems are for single gravida. Multi-bed and multi gravidafetal monitoring central system is centralized management and has high efficiency[5]. Considering the convenience of android device, to implement multi-bed central station fetal monitoring system client on the android device is meaningful.

II. REQUIREMENT ANALYSIS

A. System Overview

Fetal monitoring central station system include sub-machine, central station clients, and central station server. The sub-machine is ultrasonic sensor probe with communications modulated distributed in various wards which attached to gravida, mainly include B5, K9 and other models. Central Station client is the user interface, user can view the curve of fetal monitoring, manage document, control sub-machine. Central station server is the central hub of the system, responsible for data communications between client and sub-machine.

B. Client Functional Requirements

The client is the user interface that multi-bed obstetrical central monitoring system provides. By using a client, user can view a list of all cared gravida beds in the left area of the screen where gravida information and current fetal monitoring parameters, including fetal heart rate and uterine pressure and warning message displayed on the bed list, are shown for each bed. By clicking bed item in bed list, user can view the corresponding detailed fetal monitoring graphic chart on the right area of the screen, including fetal heart rate curve, uterine pressure curve and fetal movement.

The interactive buttons on the bottom of graphic charts can be used to control the working status of the sub-machine of the corresponding bed, turn on or off warning, jump to the document activity or curve analysis activity.

The main function of the document activity is to view the gravida document and historical fetal monitoring curve.

Curve analysis activity is to analysis the fetal heart rate curve and the uterine of the pressure curve and get monitoring results if monitor duration time reaches a certain length.

Functional requirements of the client are as follows:

User authentication: Users log in using their own account and password.

Real-time monitoring: Real-time display of current parameters such as FHR, uterine pressure, fetal movement of the beds.

Warning: Warning information will be sent to the user after the occurrence of anomalies. Warning information includes fetal monitoring parameters, sensor misplacement warning, sensor no signal warning and sensor weak electricity warning etc.

Pregnant document: Historical monitoring documents of gravida are stored.

Curve analysis: Analysis of fetal heart rate and uterine pressure curve is analyzed to facilitate medical diagnosis.

Fetal monitoring rating: Rating fetal monitoring in accordance with the existing rules on fetal monitoring curve of gravida.

C. Client Non-Functional Requirements

1) Screen adaptation

Android devices is of a wide range of different, so that the client needsto have a good display experience in a variety of device screen. There are two general principles: The first is to make the right area, the fetal monitor curve area, to be as large as possible, so the curve will be clear for user to check. The second is to ensure that the bed list is clearly visible on the left side, and easily to click.

2) eliminate the impact of unreliability in network

Android mobile devices connect to the central server via a wireless network. Mobility of Android devices and unreliability of wireless networks cause network fluctuations. Clients need to eliminate the impact of unreliability in network.

III. REALIZATION

D. System Layers

Client system is divided into four layers according to different functions, from top to bottom they are the user interface layer, business logic layer, protocol process layer, network connectivity layer.

1. User interface layer including the user interface, user interaction, and events response.

2. Business logic layer contains the business required operations and algorithms, etc..

3. Protocol process layer is responsible for the protocol resolution and package.

4. Network connectivity layer is responsible for controlling connection to the server and reconnection after network disconnection, ensure that the byte stream is sent and received accurately.

Class diagrams of the main modules as follows:

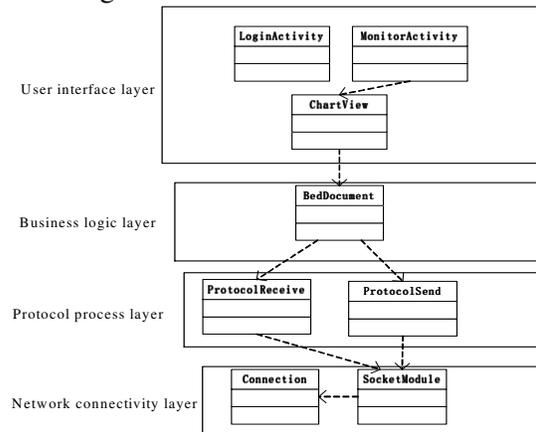


Fig. 1 system layers

User interface layer contains all the Activities. LoginActivity is login interface for user to login. MonitorActivity, the monitoring interface, contains two parts, the left part of the screen displays bed list and the right part shows information about the curve of the current bed being monitored. MonitorActivity make use of ChartView, the graph view to draw chart, which is the encapsulation of parameterized chart drawing. Its main function is to receive bed curve data, then take the device display information and display location information currently into account, and draw chart. The main Activity is show in Fig.2 to Fig.4.

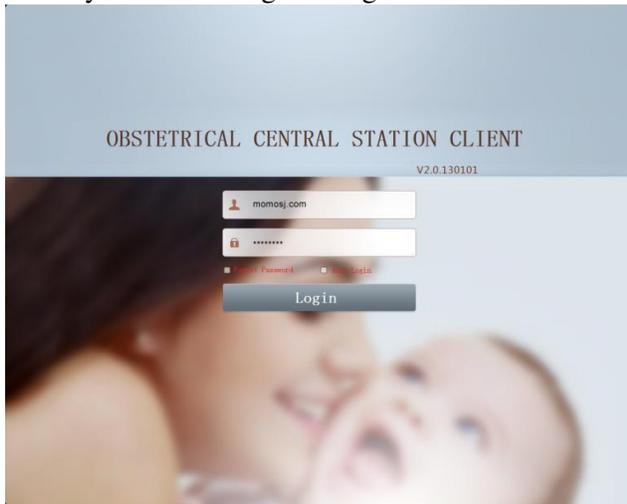


Fig. 2 Screen shot of Login Activity



Fig. 3 Screen shot of Monitor Activity

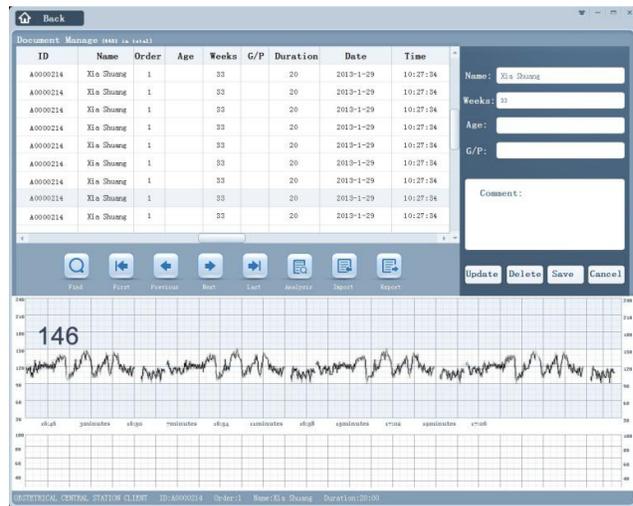


Fig. 4 Screen shot of Document Activity

Business logic layer contains document information of each bed, which is called BedDocument class. BedDocument maintains gravida personal information and fetal monitoring information while provides the basic operations of each bed. Business logic layer provide data information required by user interface layer to display , and use the data provided by the protocol process layer.

Protocol process layer is mainly responsible for the protocol resolution and protocol package. After being received by the network connection layer, an array of characters is thrown up to the ProtocolReceive class of the protocol process layer to resolve, ProtocolReceive resolve an array of characters to get message content in accordance with protocol agreement. When the Business logic layer needs to send data to the server, ProtocolSend class of the protocol process layer will be called to convert the data to an array of characters in line with protocol and then call network connectivity layer to send.

Network connectivity layer's main function is to manage the network connection, ensure that the client and server get connected, and to ensure communication between the two. The class Connection is responsible for ensuring connectivity between client and server. By appointing an agreement that server send heartbeat in a fixed time interval when the server has no message to send to client, the client can determine whether the connection is fail. Thus, wireless connection

unreliability problem is solved. SocketModule class is responsible for managing Socket communications between the client and the server. Through the check of the packet length, the problem of data transmitting loss that may occur can be solved .

E. Communication Protocol

Communication protocol defined in application layer of TCP / IP protocol, so the stability and reliability of communication can be ensured by TCP, the reliable communication protocol. Server respond to multiple clients simultaneously, and provide all the data of fetal monitoring and forward control signal to sub-machine.

Communication protocols are in the form of head and tail bound with special character filled[6]. Protocol begin with four consecutive 0xFD, and end with two consecutive 0xFF. The message content is in the middle .

TABLE I
FORM OF MESSAGE

Header	Contents	Sub-message	Sub-message	Tail Message
--------	----------	-------------	-------------	-------	--------------

The sub-messages in the communication protocol are in the form of character counting template[6]. The first bit of the sub-message indicates the type of sub-message, and the second bit of the sub-message indicate the length N of the sub-message. The following N bits immediately are the sub-message content. Sub-message format is as follows:

TABLE III
FORM OF SUB-MESSAGE

Message Type	Message Length	Message Content	Message Content
--------------	----------------	-----------------	-----------------	-------

Message types include two categories:

(1) Server to client:

TABLE IIIII
PROTOCOL FROM SERVER TO CLIENT

Constant Name	Constant Value	Description
SC_HEART_DATA	0x00	Heartbeat Packets
SC_BED_DATA	0x01	Bed Data
SC_NET_ALARM_DATA	0x02	Network Warning Data
SC_BED_WARN_DATA	0x03	Bed Warning Data
SC_BED_ONLINE_COMMAND	0x71	Bed Online Command

SC_ prefix indicates the message is sent from the server to the client.

SC_HEART_DATA represents the heartbeat packet. When there is no data to send to the client, the server sends heartbeats to each connected client repeatedly in a fixed time interval, providing the client the signal to check whether the connection is failed, and to prevent the client from confusing no communication data with disconnection happening. Client will consider connection has disconnected if neither heartbeat nor any message data received over a certain time interval.

SC_BED_DATA represents bed data, including fetal heart rate data, the uterine pressure data and fetal movement data, etc..

SC_NET_ALARM_DATA indicates the network warning data which alert such hardware problem as low power or network problem of the wireless probe in gravidia.

SC_BED_WARN_DATA indicates bed warning data, which means abnormal information warning is found in fetal monitoring parameter.

SC_BED_ONLINE_COMMAND indicates bed online status, power state of sub-machine of the bed corresponding to online or offline state.

(2) Client to server:

TABLE IVV
PROTOCOL FROM CLIENT TO SERVER

Constant Name	Constant Value	Description
CS_CLIENT_DATA	0x01	Client Data
CS_DOCINFO_DATA	0x02	Monitor Window Data
CS_DOCUMENT_DATA	0x03	Document File
CS_DOCMANAGE_DATA	0x04	Document Management
CS_USERMANAGER_DATA	0x05	User Management Module
CS_DIC_DATA	0x06	Dictionary Set Module Command
CS_REPORT_DATA	0x07	Report Manager Module Command
CS_REPORTSHOW_DATA	0x08	Report View Module Command

CS_ prefix indicates the message is sent from the client to the server.

CS_CLIENT_DATA indicates that the device information data of the client.

CS_DOCINFO_DATA indicates monitoring window information data, including the number of windows, screen size.

CS_DOCUMENT_DATA indicates the message is document information submitted by the user.

CS_DOCMANAGE_DATA indicates the message is document management information, including create, read, update and delete.

CS_USERMANAGER_DATA indicates the message is user management information, including create, read, update and delete.

CS_DIC_DATA indicates the message is system dictionary, including create, read, update and delete.

CS_REPORT_DATA indicates the message is report, including create, read, update and delete.

CS_REPORTSHOW_DATA indicates the message is to show reports.

Sub-messages sent by the client to the server involved in too much detail, omitted here.

F. Screen Adaptation

Due to the openness of Android system, there are a wide range of equipments. To adapt to the different screen size and resolution, Android devices provides a variety of different ways, including 9.png picture, different resolution drawable resource, layout parameters weight and others[7-9]. Screen adaptation of central fetal monitoring station client needs a variety of techniques to fulfill the requirements. The main method is the use of parameterized layout.

When the client is running at first time, Android API is called to obtain the screen size and resolution information of the device. When the device screen size is less than 5 inches, the weight parameter in the layout is used to set layout; When the device screen size is greater than 5 inches, fixed dpi is used so that the actual width of the bed list on the left side of the display screen is a fixed value. This ensures that the bed list is not becoming too wide and occupy too much display space as the screen size of the device change to larger, thus make as more as area for the right side of the screen to display fetal monitoring chart graph which needs more area to make its detail clearly shown.

Specifically, when starting up, the client check the property named First_Run of the sharedPreferences, which stores whether the client has run, if the First_Run is false or not exist, the device screen size and resolution will be check, and then the First_Run will be set to false.

If the screen size is smaller than 5 inches, the client uses the first set of layout, which is similar to the composite layout mode [10]. First, the entire screen is divided into two LinearLayouts, in a same parent LinearLayout. The two LinearLayout are set the weight property which makes their width ratio is 3 to 7. Internal elements of the two LinearLayout are filled in the entire parent layout, which ensures them have a fixed display ratio of 3 to 7.

If the screen size is larger than 5 inches, the client uses the second layout. First the relationship between dpi, pixel and the length of the actual display is found out as follows:

$$px = dp \times (dpi / 160)$$

$$dpi = \frac{\sqrt{px_L^2 + px_W^2}}{L_{Dia}}$$

The current Android devices have fixed screen ratio 16: 9 basically, that

$$px_L : px_W = 16 : 9$$

After substitution we can get:

$$dpi = \frac{\sqrt{px_L^2 + \left(\frac{9}{16} px_L\right)^2}}{\frac{\sqrt{16^2 + 9^2}}{16} L_L}$$

After substitution we can get:

$$dp = 160L_L$$

The dp described above can be used to ensure that the same user interface element has the same actual display length on different device screen size.

The desired bed list width in the left area to display is 2 inches. We take LinearLayout width to be 320dp. So that, for all the device larger than 5 inches, the display width of the bed list in the left side is always fixed on 2 inches. Thus not only do we ensure the bed list clearly visible and easy to click, but also ensure the bed list will not increase the width as the screen size change to larger, occupy too much area so that decrease the width of the chart view. When using a larger screen equipment, width of fetal monitoring chart graph on the right will be wider, benefit to the user a clear view of curve.

G. Parametric partial drawing

Fetal monitoring chart graph on the right part of the screen take pixel as the unit to draw. Thus the

client can make the paint graphics display better. but different devices are in different resolutions. we need parametric drawing. Before drawing the client first set the parameters of the drawing area, including the width and height pixel values, and the pixels per point of the curve. The parameters above passed to the constructor of the drawing class ChartView. The constructor initializes using these parameters. When drawing the curve, the line width of curve depends on the pixels per point parameter.

When drawing the curve, if the entire curve data is put into the image view, it may run out of memory. In this case, the curve belongs to the view element that has less effective pixel but quite wide view width, so we consider overriding ondraw() method of view, and only draw within the screen scope, thus we can effectively save memory. Since the curve has less effective pixels, such an approach needs not too much workload and has feasibility. When dragging the curve, we need to pass the new curve location value from onTouch() method to ondraw() method to redraw, which make sure that then curve displays properly after dragging.

By recording the beginning curve point index number at the left side of the screen, we can get the time of the point on the left side at the beginning of the shown curve by calculating, then we sequentially plot to the right until the right end. During the plotting process, when the corresponding time of the current curve point is on 00 second, a vertical solid line should be drawn; when the current time point is on 20 or 40 second, a vertical dotted line should be drawn. This makes user easier to acquire time of the curve point. When dragging the curve, only we need to do is to update the index number of the leftmost start point and redraw.

H. Disconnection Reconnection

Android mobile devices access the internet via a wireless network generally. Many uncertainties exist in wireless network [11]:

- Android device wireless network is not turned on;
- Android device wireless network is turned on but do not select a wireless router;

- Wireless router connection is failed;
- Network access is prevented by the filter rules of the wireless router;
- Wireless Router is not connected to the network;
- Switching to a different router leads to different subnet thus the central station server can not be connected;
- Server connection failed;
- Server Busy;

Android device wireless network is not turned on;
Android device wireless network is turned on but do not select a wireless router;

Wireless router connection is failed;

Network access is prevented by the filter rules of the wireless router;

Wireless Router is not connected to the network;

Switching to a different router leads to different subnet thus the central station server can not be connected;

Server connection failed;

Server Busy;

Confirming the above uncertainties term by term needs to spend a lot of time and system resources, and there could be some unknown problems exist. Considering the final results, as long as the client can receive the signal sent by the server, it can be considered that the network is connected, or that the network connection is failed. In order to distinguish no message form the server between network connection failure, the heartbeat signal is introduced. When the mobile client and the central station server have connected to, the server periodically send a heartbeat signal for confirming the connection even if there is no message to send to the client. The client will consider the network is disconnected if it has not received any heartbeat signal or other message over a certain time and notify the user, trigger reconnection mechanism, try to connect to the network.

IV. CONCLUSIONS

Obstetrical central station client based on Android platform. The communication protocol defines the needed interfaces to implement function of multi-bed central fetal monitoring system, which meet the needs and have scalability. By using two sets of layout plan and parameterized partial drawing the Android device screen adaptation is completed and improve user interface experience. Heartbeat mechanism is a simple and effective solution of various type of server connection failure. The implementation complete the product line, which won the company widely attention in an exhibition of a medical and become the focus of the company's future development.

ACKNOWLEDGMENT

We would like to express our gratitude to all those who helped us to complete this work. We want to thank Mr. Lu Raosheng for his continuous help and generous assistance. He allowed us to use the resource of the SUNRAY MEDICAL APPARATUS CO.,LTD.

We have furthermore to thank Mr.Zhou Jipeng, Head of the Department of Computer Science, to encourage us to go ahead and provide laboratories for us.

REFERENCES

- [1] Zhao yongrui,Wang huijin. (2013)Analysis and Design of Multi-bed Fetal Monitoring System. [Online].Available:Sciecepaper Online:<http://www.paper.edu.cn/releasepaper/content/201307-359>.
- [2] Gavalas D, Economou D. "Development Platforms for Mobile Applications: Status and Trends," IEEE Software, vol. 28, pp. 77-86, 2010.
- [3] Sun ying, Song jian. "Remote ECG Detection System Based On Android," Computer Applications and Software, vol. 30, pp. 326-329, Apr. 2013.
- [4] Korostelev M, Bai L, Wu J, et al. "Body sensor networks in fetal monitoring with NFC enabled Android devices," Proceedings of the 7th International Conference on Body Area Networks. ICST (Institute for Computer Sciences, Social-

Informatics and Telecommunications Engineering), 2012, p. 9-12.

- [5] Lu raosheng, Wang huijin, Liu guangchang. "SRF618S Central Monitoring System in Obstetrics," Beijing Biomedical Engineering, vol. 20, pp. 85-87, Jun. 2001.
- [6] Zhao yongrui. "The Design and Implementation of a Maternal-Fetal Central Monitoring Network System based on C/S and B/S Architecture,"M. Chn. Thesis, Jinan University, China, Jun. 2014.
- [7] Cao qingchun, Li jing. "On Android Screen Self-Adaption," Discovering Value, vol. 3, pp. 63-63, Mar. 2013.
- [8] (2015)Supporting Multiple Screens [Online]. Available: http://developer.android.com/guide/practices/screens_support.html
- [9] Phillips B, Hardy B. *Android Programming: The Big Nerd Ranch Guide*. Pearson Education, 2013.
- [10] Nudelman G. *Android design patterns: interaction design solutions for developers*. John Wiley & Sons, 2013.
- [11] (2015)Connecting to the Network [Online]. Available: <http://developer.android.com/training/basics/network-ops/connecting.html>