The hardware and software used for this project was a gift from IEEE Computer Society to IEEE Students' Branch, Institute of Technology BHU, Varanasi, India. This project competed for the Computer Society International Design Contest 2000 and was placed among the TOP 20 Projects

DOC@HOME: A Health Care System

Dr S. C. Gupta
s.c.gupta@ieee.org

Hridesh Rajan
hridesh@ieee.org

ABSTRACT

DOC@HOME is a facility provider, which exploits the availability of the cheap, reliable computational power as well as the ubiquitous communication facility, the Internet. By integrating these evolving technologies and enabling interaction with the available expertise in the medical field, it provides easy accessibility to a user with a pool of medical experts available anywhere anytime. It has two main components: information Appliance (iA) and Medical Service Provider's Server (MSPS). The iA is intended to be used in a home to provide interface with MSPS through a low bandwidth wireless link. The MSPS, connected to the internet, maintains all the databases, for the iAs and the doctors, and provides an active platform for communication between the two. It filters the vast amount of information available through internet for specific relevance to the two set of users, and presents in the desired format. There is a provision for online monitoring of temperature, respiration and ECG. Respiratory signals are used to detect and alert for apnea. The prescription of the doctor is also communicated to a chemist nearby the patient. The patient is alerted for taking the specified medicines at the prescribed time. When a user seeks first aid information, the relevant details are supplied by the MSPS. In emergency situations, when a user presses a PANIC button, doctors and friends/ relatives are alerted. The information Appliance is designed to be compact, lightweight, rugged, and highly reliable. It is ergonomically designed for user convenience, and has features for saving the battery power. Especial provisions have been added for the aged, and modifications proposed for the physically challenged.

1. Introduction

Even in this age of information technology priviledge of access to a medical expert anytime anywhere is not an achievable fate. Most of the time in critical situation accessibility to a medical expert not only helps but proves to be a lifesaver. Not only in critical situations, but sometimes in relaxed hours also availability of an expert advice eases the situation.

Most of the methods to facilitate online expert advice and emergency help rely on the human being to communicate the matter to appropriate place. If the person in need is not capable of communicating his/her condition which can be either due to unconsciousness (Heart Attack /accident) or the person being ignorant of the appropriate place to contact (New to the place/suffering from amnesia) it mostly results in fatal cases. This is precisely the place where `DOC@HOME` fits in.

Rest of the documents is organized into 7 sections. Section 2 gives an overview of the system. Section 3 lists the performance requirement considered for the system. Section 4 describes the design methodology used. Section 5 details the Implementation and engineering considerations. Section 6 discusses the tradeoffs considered during the design process. Section 7 describes how the testing was done and Section 8 concludes.

2. Overview

The system DOC@HOME has been designed to provide medical assistance to a user remotely. It is centered around a Medical Service Provider, which maintains several interconnected Servers at different locations. Each server provides medical assistance to a set of users, allowing accessibility through an information appliance of the user. The server maintains database of all the users connected to it, monitors health status through sensors, and provides specific guidance, including first aid instructions. It is expected to have stored database of available medical experts and can help identify a suitable expert for a particular patient. It presents a desired view of the database of the patient to the doctor and provides an active platform for communication between the two. It can also establish links with appropriate information sites available over the Internet, and filters it for easy and effective usage of the doctor and the patient.

The user interface, information Appliance, is built on a small, dedicated single board computer system, VL-686-2 a product of VersaLogic Corporation. Add on to the SBC are 8- MB flashdisk DISKONCHIP a product of M-systems, 260 MB PCMCIA IDE called Callunacard a product of Calluna technology, three sensors: for temperature, respiration and ECG, and their signal processing circuits. While the memory available in the system through the DISKONCHIP and the Calluna card will be sufficient, the communication bandwidth available via a low bandwidth wireless link (9.6 KBPS) is expected to fulfill the requirement.

The Medical Service Providers Server (MSPS) can be ported on any commercially available POSIX compliant computer system. We have found the real time environment of the QNX operating system most suitable for developing the communication and control related programs of the information Appliance. For implementation of databases, we have used the Pervasive SQL.

3. Performance Requirements

Some of the performance requirements were:

- The system should provide secure means of authentication.
- The sensors must give proper output within the temperature range [10-50] degree Celsius.
- Communication between the iA and server should be such that the data rate does not increases the 9.6 KBPS limit set for a wireless link. (As per the GSM standards).
- The whole program along with its data should fit in the Disk On Chip provided by the M-Sys and Calluna IDE card.
- The system must go to a low power standby mode if no activity is going on for 5 minutes.
- The time required to respond to a single query must not be greater than 5 seconds.
- Preprocessing of the database should be interleaved between queries such that the query time is not affected by it.
- The server should automatically record date and time on which a data is recorded.
- The system must support I/O devices like keyboard mouse.

4. Design methodology

For designing MSPS, different Objects were identified, and interaction among them was defined. Flow chart of the Web - Interface was made for designing the WWW interface for the doctor. For hardware design of the iA, block diagram was made and then every block was further divided until we reached the point where all the components were either available with the kit or could be bought off - the - shelf. For its software design, an Algorithmic State Machine (ASM) diagram of the system was first made, which was then used to determine the state of various objects and changes in them through out the execution. The development process chosen by us for this project is iterative enhancement model, which counters the limitation of the waterfall model and tries to combine the benefits of both prototyping and the waterfall model. The basic idea is that the software should be developed in increments, each increment adding some functional capabilities to the system until the full system is implemented. Accordingly, each of the ASM and Block diagrams was incrementally built, and analyzed by us. Looking into the system from different views: control view by ASM and system/object view by block diagram, helped us in making quick and precise the specifications of the modules and interface amongst them. Also, this approach resulted in better testing, because testing each increment is easier than testing the whole system as would have been the case in the waterfall model.

5. Implementation and engineering considerations

An overview of the system DOC@HOME is shown in Fig.1. The sensors are connected through the sensor circuitry to the inbuilt ADC through the port j10 of the SBC. Panic button was attached to the SBC through the digital input port. The sensor ready and appliance ready are the two status outputs through the digital output port. The details of the sensors and application to

The hardware and software used for this project was a gift from IEEE Computer Society to IEEE Students' Branch, Institute of Technology BHU, Varanasi, India. This project competed for the Computer Society International Design Contest 2000 and was placed among the TOP 20 Projects

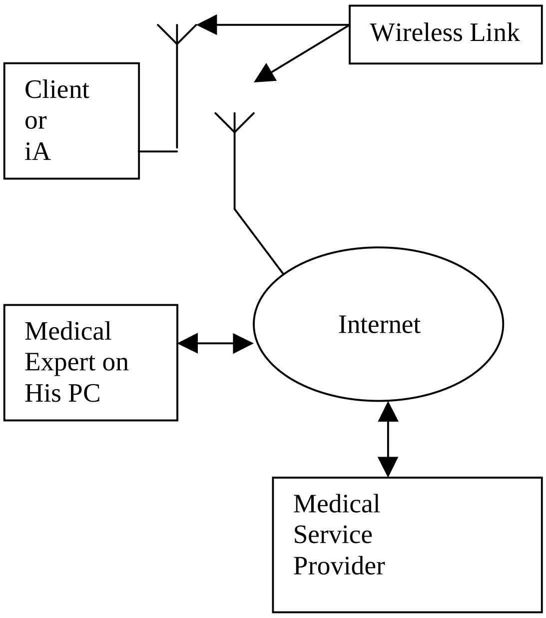

Fig. 1 Overview of the DOC@HOME System

which they are put, interface and role of the PANIC button are described in the following sections

5.1 Sensors

The appliance has facility to measure vital signs of a patient using a set of external sensors. Presently, provision has been made for three sensors: ECG, respiration and Temperature. The information regarding the sensor type and their calibration is written in a sensor initialization file. Every time the system starts, the information is read from this file. Additional sensors can easily be added to the system by writing their calibration and type information in the initialization file. For connecting the sensors to the body, necessary instructions are sent by the server, if desired by the patient. Their output is connected to the appliance through a socket interface capable of expansion for up to 8 sensors. Four terminals are required for every sensor: one for the analog input, and three for the power supply (+15V,- 15V and GND) The analog inputs are connected to channel 0-7 of the 12-bit A/D converter through connector J10 of the single board computer provided with the kit. The analog inputs are in --5 V to + 5 V range. To sample the analog input the analog control register at 00E4h is set to x=18h. After the conversion is complete the data is read from the ADCLO and ADCHI registers at 00E4&00E5 respectively. The signal value, in Volts, is obtained from the corresponding digital value by multiplying it by the step size (0.004882813). When the sensors are not being used, the A/D converter is powered down by setting the Analog Control Register (ACR) to C0h (Full power down mode to save the battery power).

5.2 Temperature

Temperature of the patient is send to the server at the predetermined regular interval. or when the variation between two successive observations is more than 0.5-Degree Centigrade. The server records the temperature along with the time it was measured. Doctor can visualize the variations in a graphical form.

5.3 Apnea

The iA locally monitors the respiratory signal for possible apnea. If and when detected, it raises an alarm locally and sends an emergency signal to the server.

5.4 ECG:

At present ECG signals are monitored through a single probe. The ADC samples the signal at the Nyquest rate. After Compression, the digital data is sent to the server for maintaining its record. For analysis by doctor, the data, after decompression is presented in usual graphical form along with the suitable analytical parameters.

5.5 PANIC Button

When the kit starts up, it spawns a highest priority process to take care of the PANIC button. When the panic button is pressed this process is immediately scheduled. It results in contacting a doctor, presenting him the case study of the patient and the sensor signals online. Messages are also sent to the relatives and friends of the patient.

The hardware and software used for this project was a gift from [IEEE Computer Society](#) to [IEEE Students' Branch, Institute of Technology BHU](#), Varanasi, India. This project competed for the Computer Society International Design Contest 2000 and was placed among the TOP 20 Projects

5.6 Data Objects of the System

Specification for the object oriented design for the MSPS system.

Class Symptom_Data
Sysmtom_no
Symtom_name
Symtom_value
Symtom_priority_factor
Symtom_date
Symtom_duration
Symptom_location

Symptom_no is a unique identity for a symptom. Symtom_name tells the name of the symptom.Symtom_value is a numeric value of the symptom like blood pressure etc. Symtom_priority_factor tells how much is the abnormality in the symptom and how much important it is for the detection of disease. When a doctor is analyzing the patient's records. They are organizes by the priority factor. Symtom_date is the date and time when the readings were taken. Symtom_duration is the time from which the patient is experiencing the symptom. Symptom_location tells the location of the symptom. There are some words, which are to be used while describing location. When Querying the data these words are necessary to search the right spot.

Class medical_info
Serial_no
Height
Weight
Sex
Temp
Blood_pressure
Heart_rate
Pulse_rate

Class history
Serial_no
Test_results
History_present_illness
History_previous_history_illness
History_menstrual_history
Treatment_history
Family_history
Social_history
Occupational_history

Class Medicine
Ingredients
Name
Disease
Extra_effect
Currency price
Producer
Customer_care_no

Class Disease
Name;
Symptom_Data
Cure
Precautions
Medicine
Expert

5.7 Communication Protocol

The communication protocol between the server and the iA is illustrated via a mock session proceedings of which are detailed in the followings. In this, S - and C precede the responses of server and iA -- respectively. It is assumed that the iA has been switched on and a connection with the MSPS has been established.

S - welcome to DOC@HOME /* the welcome message of the MSPS */
C -- login = login-name /* Client provides the login name */
C -- password = password /*In encrypted form using the public key algorithm.*/
S - 110:connected /* if the login is successful 110 and 210 are the codes for success and failure of the login procedure/
210:login incorrect. Connection closed./* If the login is unsuccessful */
C -history: /* As a result of any command (Say history) on the iA, it sends following message to the MSPS.*/
S - Sending History. terminated by <cr>.<cr>/* to detect the start and return of the message the sequence in the message is stuffed by <cr>.<cr>*/
S -History of the patient currently logged in
S -< cr>.<cr>/* End of the first history session/
C -Received __ characters/ *feedback of the iA*/
S - OK resending history./*Start of the second session of history */
C - Sendmail: /* As a result of command Sendmail on the iA, it sends following message to the MSPS.*/
S -From: /* asks for the source address */

C - X@yz.abc/* Source address provided by the iA */
S -To :/*Asks for the destination address */
C - M@nx.abc/* destination address provided by the iA */
S -CC :/* Asks for the carbon copy addresses*?
C - P@qrs@abc/* carbon copy addresses provided */
S -Text :/* Asks for the text of the message */
C -Text of the mail /* text of the message send by the iA */
C -<cr> .<cr>/*text terminated */
S - Received __ characters/ *feedback from the MSPS*/

6. TRADEOFFS CONSIDERED DURING THE DESIGN PROCESS

6.1 Whether doctor would communicate directly to the iA or through the MSPS?

Communicating directly will save a lot of overhead of communication, but for saving this overhead, complex authentication algorithms must be executed on low end iA. Direct communication between the doctor and iA will require that either the iA should run a web server or doctor end should have a program to communicate with the iA. Running a web server on the iA is infeasible and having a program on the doctor's end bounds him to the PC on which the program is loaded. In case of several doctor's communicating with the client at the same time or asking for some client's data at the same time handling will be difficult at the iA. Whereas at the MSP's server end a full fledged database application can be run. The program to be run on the doctor's end could be a Java applet to be downloaded from the DOC@HOME Web-Site but it will be a source of security potholes. Further most of the time queries of doctor will not be from the patient but from his database. So directly communicating with the iA is useless.

6.2 Database of the clients

Database of the clients can either be kept on the iA itself or centrally at the MSP's server. Keeping the database at the iA will keep communication overhead at the minimum, but if suppose the person has somehow misplaces his iA all his information is lost. Keeping the database completely at the server will make the iA independent of person. If the person having a iA at his home visits another person having a iA he need not require to carry his iA with him all the time. If suppose at the friends home he needs the iA he can simply login on his friends iA with his login-id and the iA works as if it is his own.
Most of the doctor's queries will be from the client's database and since doctor will communicating through the server it is better to put the database there to avoid the congestion on the low bandwidth wireless link between the server and the iA.

6.3 First aid instruction pages

There were three options before us:[1]to keep the pages entirely on the server,[2]to keep the pages entirely on the iA and [3]to keep the pages partly on the server and partly on the iA. The first approach is rejected on the basis that most of the time first aid pages will be accessed in emergency situations. Any congestion between client and server may lead to delay in page display and this will make the situation more panicky. The second approach is rejected on the basis that this will need a huge storage space on the iA adding a cost on it. The third approach keeping initial few pages on the iA and rest on the server, serves both the purposes when the initial pages are demanded those are presented immediately and with some branch prediction further pages are imported from the server.

6.4 Sensors

Sensors are kept as external removable devices. This is done to keep flexibility in the load to be carried by a patient who is constantly monitored. There is no point for a person to carry a respiratory sensor when he/she is being monitored for heart attack and vice-versa.

6.4 Operating system

QNX: Initially we were quite apprehensive about this product due to unfamiliarity with it,no experience of using the Photon Application Builder and lack of documentation, but after all it proved to be a gem. The real time capabilities of the system was the added benefits. The EVENT DRIVEN nature of PHAB really helped us in converting our Algorithmic State Diagram to real system in no time. The least the demonstration floppy with this system helped us in planning the executables and data files for our system to a minimum.

The hardware and software used for this project was a gift from IEEE Computer Society to IEEE Students' Branch, Institute of Technology BHU, Varanasi, India. This project competed for the Computer Society International Design Contest 2000 and was placed among the TOP 20 Projects

7. Verification and testing:

The values selected for testing of the modules were:

- Some of the logical inputs.
- For each of the input maximum and minimum values.
- Inputs such that the output takes maximum and minimum values.
- Inputs selected such that each of the branches of the module is traversed.

Some of the Modules like login, checkup and panic were tested exhaustively to ensure high reliabilty.

For testing of communication between the iA and the MSPS illegal strings were generated in response to the proper requests from iA. Long strings were send in response such that the buffer overflows at the iA end and Connection was closed at each of the states of the iA-MSPS communication .For testing of the MSPS illegal commands including long strings were generated, proper commands with invalid parameters (e.g. after sending the command hello doctor instead of doctor's address an integer was send, after sending the command log temperature ECG data was sent instead of the temperature) were given as test cases , connection was closed at each of the states of the iA-MSPS communication and many connections were simultaneously made to the same MSPS. For testing of the iA-MSPS system we checked for the cumulative data rate between the MSPS and the iA(Performance requirement states that it should not exceed 9.6 KBPS for a wireless link).

For testing the Database part some of the quantitative measure like query time, response time preprocessing time and update time were tested.

8. Conclusion

This paper describes the implementation of DOC@HOME. In the future videoconferencing, voice communication with doctor or any other iA can be added. Addition of more sensors to the machine is possible. Automatic emergency handling (e.g. automatic injection mechanism), Automatic positioning of the iA with the help of Global Positioning system to rush help to the patient in case of emergency (Apnea attack, heart attack Accidents etc.) will also be considered. Provision for looking after the financial part of the medical treatment like insurance etc. are being considered. Additional support for disabled like a Braille keyboard and screen for blind people can also be done. Voice to text and text to voice conversion facility to allow a deaf to talk to his/her doctor is also a viable option. Depending on the advancements in the Field of AI an expert system to assist the human expert can also be considered.

Reference

[1] Furht, B, Perez, A: Adaptive Real Time ECG compression algorithm with variable threshold, IEEE transactions on biomedical Eng. pp. 491-493,Vol. 35, No.6, June 1988.

[2] Socikh & Yesavage: Yesavage Geriatric Depression Scale (GDS) -- Short Form. Clin Gerontol 5:105,1986

[3] Lin,y. and Son S.H: Concurrency control in real time databases by dynamic adjustment of serialization order. Proc. IEEE real time symp.pp.104-112

[4] Michael S.: Huchinson's Clinical Methods, nineteenth edition Bailliere Tindall 1989 London.

[5] Tanenbaum A.S.: Computer Networks, Prentice-Hall Englewood cliffs NJ. 1991

[6] Comer D.E: TCP/IP vol. III, Prentice-Hall Englewood cliffs NJ.

[7] Kerten R.: Getting started with QNX4, a guide for real time programmers Parse Software Devices.

[8] Krishna and Shin: Real time systems, McGraw Hill

[9] http://www.qnx.com

[10] http://www.m-sys.com

[11] http://www.VersaLogic.com