Healthcare Data Fusion and Presentation using Service- Oriented Architecture (SOA) Orchestration Mechanism

Veeramuthu Venkatesh
School of Computer Science & Engineering
SASTRA University
Thanjavur

Pethuru Raj
Sify Software Ltd
Tidel Park,
Taramani, Adyar,
Chennai

Kaushik Gopalan
School of Computer Science & Engineering
SASTRA University
Thanjavur

Rajeev.T
School of Computer Science & Engineering
SASTRA University
Thanjavur

ABSTRACT

Sensors are the eyes and ears of next-generation IT. Electronic devices and smart artifacts (Ordinary objects and materials are digitalized and empowered to be networkable , sensing, sensitive and computational) are being increasingly interlinked to understand the contextual information perfectly and precisely in order to cognitively ponder about the course of actions and to implement the tasks identified dynamically in real-time. There are several domains yearning for such kinds of enablement, seamless connectivity, spontaneous integration, adaptive composition, and cognitive collaboration among different, decentralized and distributed everyday entities and elements in our daily works and walks. Smart Healthcare is the leading topic for research and development.

There are several kinds of physical illnesses emerging and evolving. And hence medical professionals are striving hard and stretching further to come out with intelligent healthcare processes, practices and products in order to arrest the spread of diseases and viruses amongst people. Robust and resilient medical instruments, standards, and technologies are being unearthed in order to guarantee better care and cure for humans.

With the maturity of multifaceted technologies, smart healthcare vision is on the way to fulfillment. In this paper, we have identified one pervasive and persuasive use case. One such case is implementing a smart environment in a hospital ICU where even a small time difference in treatment can change course of the entire medical procedure. The smart environment is implemented using technologies like zigbee (a wireless communication protocol), OSGi (a middleware) and knoplerfish (a framework for implementing OSGi). .

General Terms

Dynamic Device Discovery (DDD), orchestration, Service oriented architecture, Wireless Sensor Networks (WSN).

Keywords
Knopflerfish, Ambient Assisted Living, smart home

1. INTRODUCTION

The Ambient assisted living which was originally developed for helping aged people might also be implemented for providing smart environments for other purposes[2, 4-8, 12- 14]. One such application is a smart hospital environment [2, 4, 6, 7]. The intensive care unit (ICU) is the most vulnerable part of a hospital environment. Any small aberration or digression in the normal flow of events can cost as much as a life of person. So it is only natural that the ICU should be made more sophisticated for contemporary hospital system. The sophistication can be implemented using pervasive computing which provides smart environment using sensors and wireless communication technologies [13].

In this paper the procedure implemented will help the doctor to monitor his patient admitted in the ICU assigned to him and apprehend the developments in his conditions remotely. To implement this, various outputs which have been given by the monitoring system have been integrated and sent to the orchestration server. A CCTV camera is installed in the patient end and the live feed is sent to the orchestration server. The orchestration server hosts a web service which has a distributed database [3, 10]. All the data along with the live camera feed is stored in the database and is displayed in the web service in accordance with the type of user (doctor or patient caretaker). The doctor gets the patient statistics while the patient's caretaker gets the status and prescription of the patient using wearable accelerometers [1, 9].

2. WSN ARCHITECTURE.

The architecture of the entire procedure is quite straight forward. For the purpose of understanding the concept, only three different types of ICU are considered namely the pediatric ICU, trauma ICU and the cardiac ICU. Each ICU has its own set of attributes which are compiled together and sent to the orchestration server via zigbee protocol. Each ICU is assigned a doctor of that specialty and there is duty doctor assigned to all the ICU is uneventful absence of the specialty doctor. All the data of ICUs are sent to the orchestration server where it is stored in the database along with the live camera feed from the CCTV camera. The web service hosted by the orchestration server displays all this compiled information and camera feed to the doctor [11]

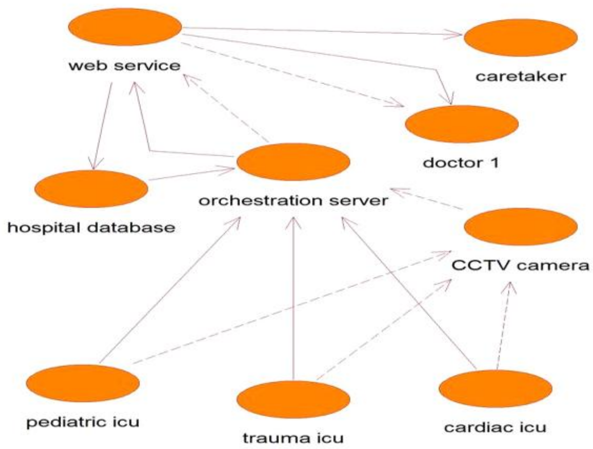

Fig 1:architecture

2.1 Scenario

The various systems that are monitoring the patient in the respective ICUs are all considered to be of different make and they all have their own input and output parameters.

2.1.1 The ICUs that are being monitored are:

Pediatric ICU with monitors for:

Blood pressure, body temperature, intracranial pressure, pulse oximeter, apnea, ECG, ETCO2 and the CCTV camera.

Neuro ICU with monitors for:

Blood pressure, body temperature, intracranial pressure, pulse oximeter, apnea, ECG, spirometer, blood glucose monitor, body weight scale, EEG, ETCO2 and the CCTV camera.

Trauma ICU with monitors for:

Blood pressure, body temperature, pulse oximeter, ECG, ETCO2 and the CCTV camera.

The java programs were created and tested first in the Eclipse IDE. This IDE was used as it has built in options for creating the activator classes and the various methods that are needed in the OSGi framework.

A method has been devised that collects all the data that is obtained while monitoring a patient in an ICU.

The data is collected after fixed intervals. Depending on the need, this interval can be changed. Each device that is being used to monitor the patient is important as only when the doctor has all the information can he understand the problem if any and treat the patient in time. The data is recorded after a fixed interval, the same way a nurse is appointed to take care of the patient. All this data was collected electronically. The data was verified at all the places.

2.2 Technologies used:

2.2.1 OSGi [4]

The proposed idea is based on the OSGI framework. The OSGi Service platform specification delivers an open, common architecture for service providers and developers to develop deploy and manage services in a coordinated fashion. It provides java framework that supports the deployment of extensible and downloadable applications known as bundles. The OSGi platform is designed in such a way that it can work without any form of management.

They can be remotely controlled by the use of a specific protocol. As long as a management system can provide a standardized OSGi protocol bundle, it can operate with any manufacturer's device regardless of the underlying protocol. The framework helps us in concatenating the data from a number of sources (which can be of different make) and use the data as required. The separate components are able to cooperate in a single

Java Virtual Machine(JVM), combine it with a live feed from a camera and send the data to the server.

2.2.2 Eclipse

Eclipse, and open source IDE for the development of the java application has been used here. The IDE has built in options that help in making projects that are meant to be deployed in an OSGi framework of different specifications. The IDE also has its own OSGi runtime environment that simulates the actual framework. It has all the basic options including the start and stop options for the deployed bundles. The IDE allows multiple projects to be opened simultaneously.

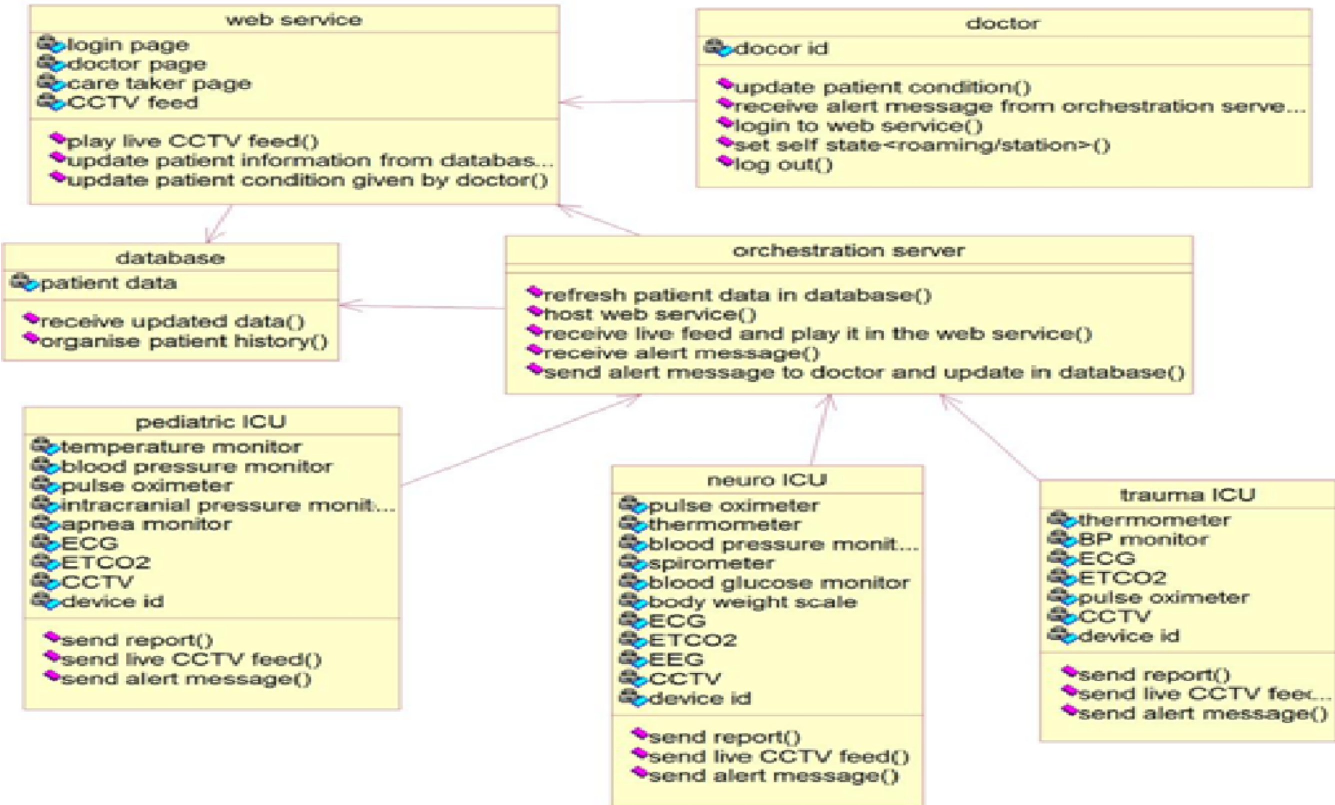

Fig 2:class diagram

2.2.3 Knopflerfish
The open source platform based Knopflerfish framework has been chosen to initialize and deploy the bundles. It is easy to install, provides a desktop user interface and is simple to use. The bundles can be opened, started and stopped at any time. The bundles can also be updated while they are deployed.

2.2.4 Apache
The open source UNIX based server, Apache was used to host the web service. Apache supports PHP, Perl, Python and Tcl. It is supported by many GUI s. To host the web service, Apache was installed in the orchestration service.

2.2.5 MySql
MySQL was used to create and manage the database. The database system handles large number of tuples and queries with no complications.

2.2.6 Joomla
The web service was done using an open source content management system called Joomla. It works with PHP and MySql. Once apache is installed, loomla can be installed over it and a web service can be created.

3. IMPLEMENTATION
There are nodes attached to each of these ICUs where the data will be obtained from all the mentioned monitoring devices. The CCTV camera footage of the person being treated in the ICU is also received by the node. The data is first collected from each device and stored temporarily in one node. This node is the client side of the communication. This node will first establish the connection with the main server. The mode of connection is the TCP/IP protocol. The protocol provides a secure connection oriented transmission. The connection oriented transmission guarantees data transmission without any loss or changes.

Random values were taken as input in the appropriate ranges. These random values were generated using an object of the Random class provided in the java.util package.

The client side will first initialize the connection with the orchestration server. The server will send back a message to confirm the connection which was displayed in the client side of the communication. The data which consists of integral and floating point numbers is converted to a string object. The concatenation features of string class were used here.

The string object is useful for all further transmissions. The final string consisting of all the values recorded was first printed in the client side so that any person monitoring the system in the ICU can also check the values there. This is done as a measure to help in identifying any form of error in the transmission or any other error that may cause the system to fail. The data is then sent to the server.

```
   client=new Socket("127.0.0.1", 2500)
   pr=new PrintWriter
(client.getOutputStream(),true);
   br=new BufferedReader(new
InputStreamReader(client.getInputStream
()));
  ir=new BufferedReader(new
InputStreamReader(System.in));
for(i=0;i<10;i++){
ss=temperature+" "+sysbp+"/"+diasbp+"
"+pulseox+" "+intercran+" "+etcot+"
"+devid+"\n";
pr.println(temperature+" " + sysbp+
"/"+diasbp+" "+pulseox+" "+intercran+"
"+etcot+" "+devid);
 }
```

Fig 3.1: Part of client side communication.

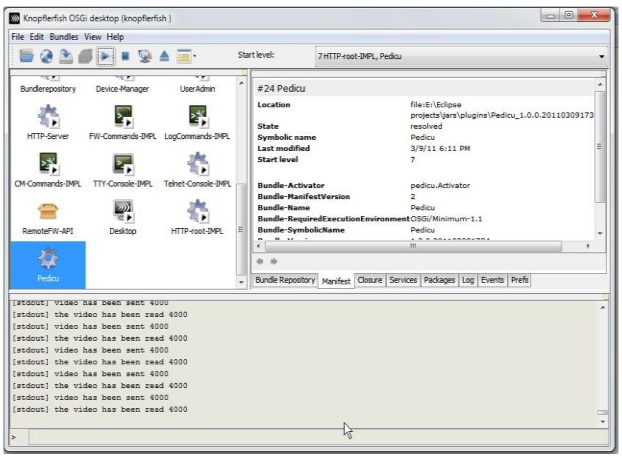

Fig 3.2: client side communication in knoplerfish window

While these data were being read from the monitoring devices, a recording from the CCTV camera was being stored in the client side. The recording is first sent to the server. The video footage is sent by another TCP connection so that the 2 transmissions (video transmission and the data from the monitoring devices) are not affected by each other and if any problem has delayed or caused and error in the transmission of one of them, it does cause any problems to the other.

```
str=br.readLine();
String data[] = str.split(" ");
for(j=0;j<data.length;j++)
System.out.println(data[j]);
try {
      UploadMySQL d = new UploadMySQL();
      d.addPedICUValues(data[0], data[1], data[2], data[3],
data[4], data[5]);
            }
catch(Exception e){
System.out.println("SQL Exception"+e);
                        }
System.out.println(str);
```

Fig 3.3: Part of code of the server side of communication.

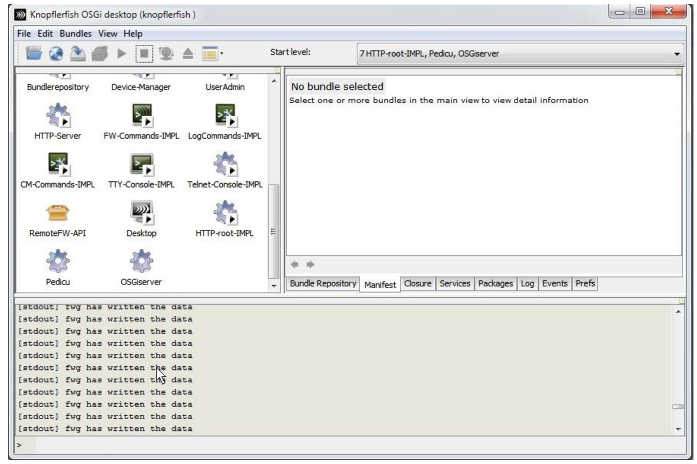

Fig 3.4: server side communication in knoplerfish window

Each ICU has one separate connection with the server. They all can simultaneously transmit data to the server. The server receives the data from the ICU and stores it in a database. The data received was a single string of input that has all the values collected from the monitoring devices. A database is maintained for the storage of this data. The database has features that help in the proper storage and retrieval of the data. Hence it was used.

```
    query = "INSERT INTO pedICU
(temp,bp,pulse,ip,apnea,etc) VALUES "
                              +
"('"+temp+"','"+bp+"','"+pulse+"','"+ip+"',
'"+apnea+"','"+etc+"')";
        stmt.executeUpdate(query);
```

Fig 3.5: part of the database program for handling the data from server. .

The database was able to read the data sent by the client as a string, extract each value and store them in the particular columns in the required format. For example, the blood pressure was sent as 2 numbers separated by a '/' and it was stored the same way. The server also stores the time at which this data was saved in the database. A simple time stamp procedure was used. The server receives the live feed of the CCTV camera in the ICU which was recorded at the time the data was collected. The live feed was also stored with the time stamped.

Initially the program was run for a small number of cycles where in each cycle, at a particular time, the entire data set from a single ICU was read. Then this data was formatted and sent to the server. The video feed was also sent to the server simultaneously. There was successful transmission of the data and the video feed. After this the program was run with an infinite loop to simulate the real scenario where the data will be collected for a certain number of days. The program ran successfully and all the values were verified in both the client and the server side.

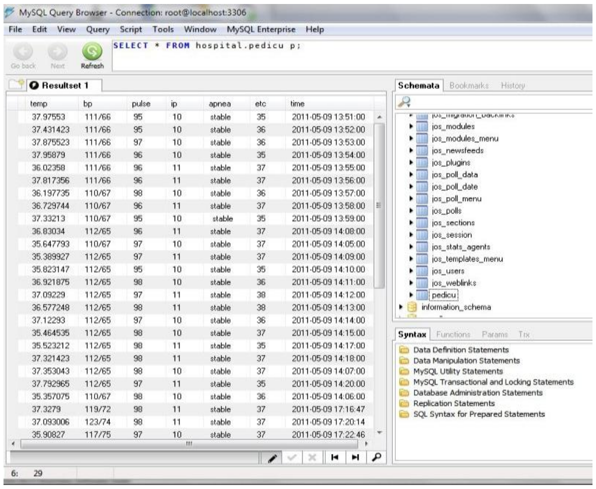

Fig 3.6: database using MySql

The web service has a login page for the doctors and one is created by the server for the patient which can be viewed and accessed by the patient's caretaker. The doctor can login to his secured login and look at the data present in the database. The data was shown in a tabular form. The patient's Id was shown. Each row of data was taken at a particular time and the time at which this data was collected is also shown. The doctor can also comment or prescribe any medication or any change to the present treatment through the comments section.

The caretaker can view the various treatments that were given and the comments given by the doctor. In the photograph, the caretaker can see the comment" the patient's condition is stable" which was posted by the doctor. Using SQL queries, we can easily view specific columns or the maximum, minimum values or any other specific query posted by the doctor. This can be useful in determining the specific changes that have occurred in the patient and can be used by doctors if needed.

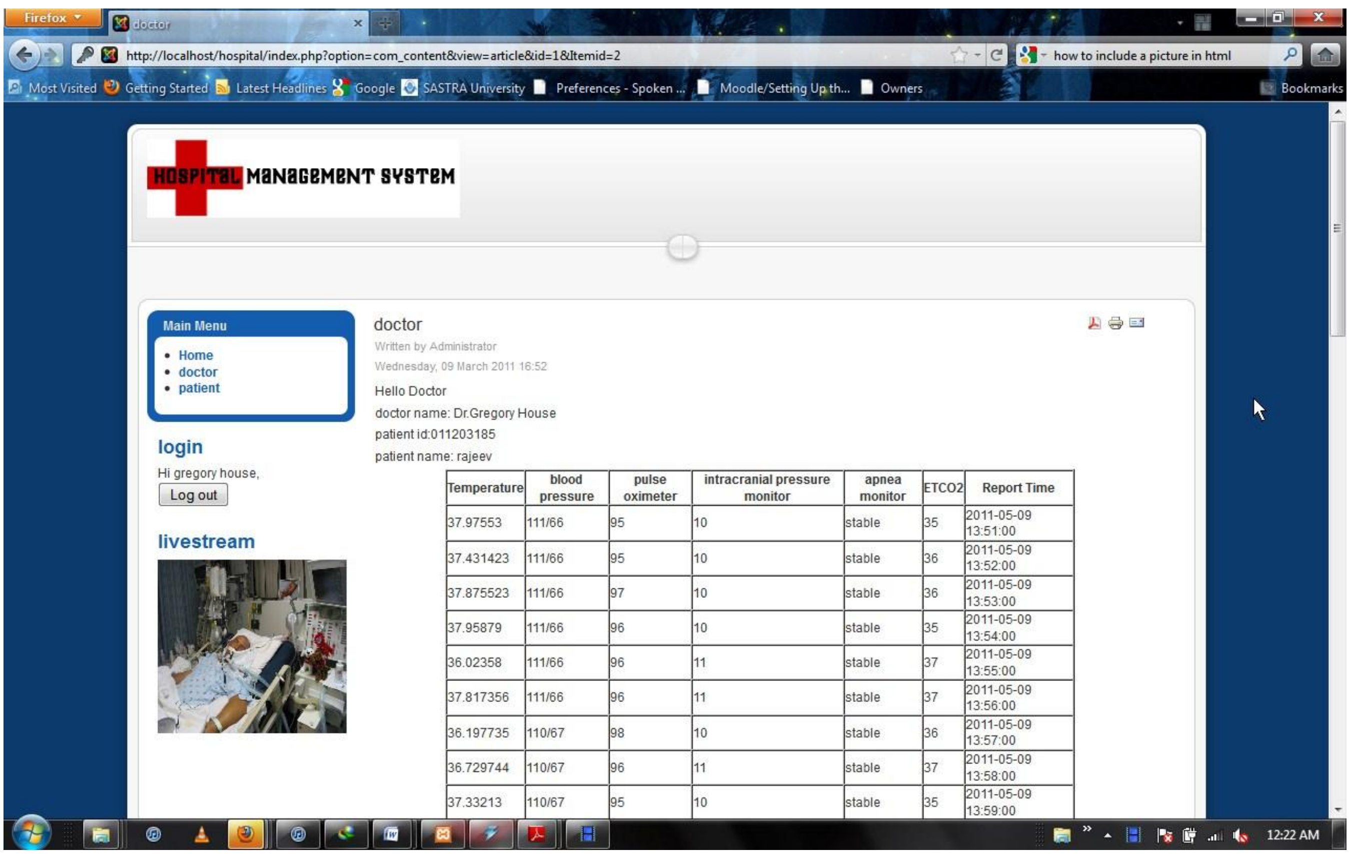

Fig 3.7: Server Database for Doctors.

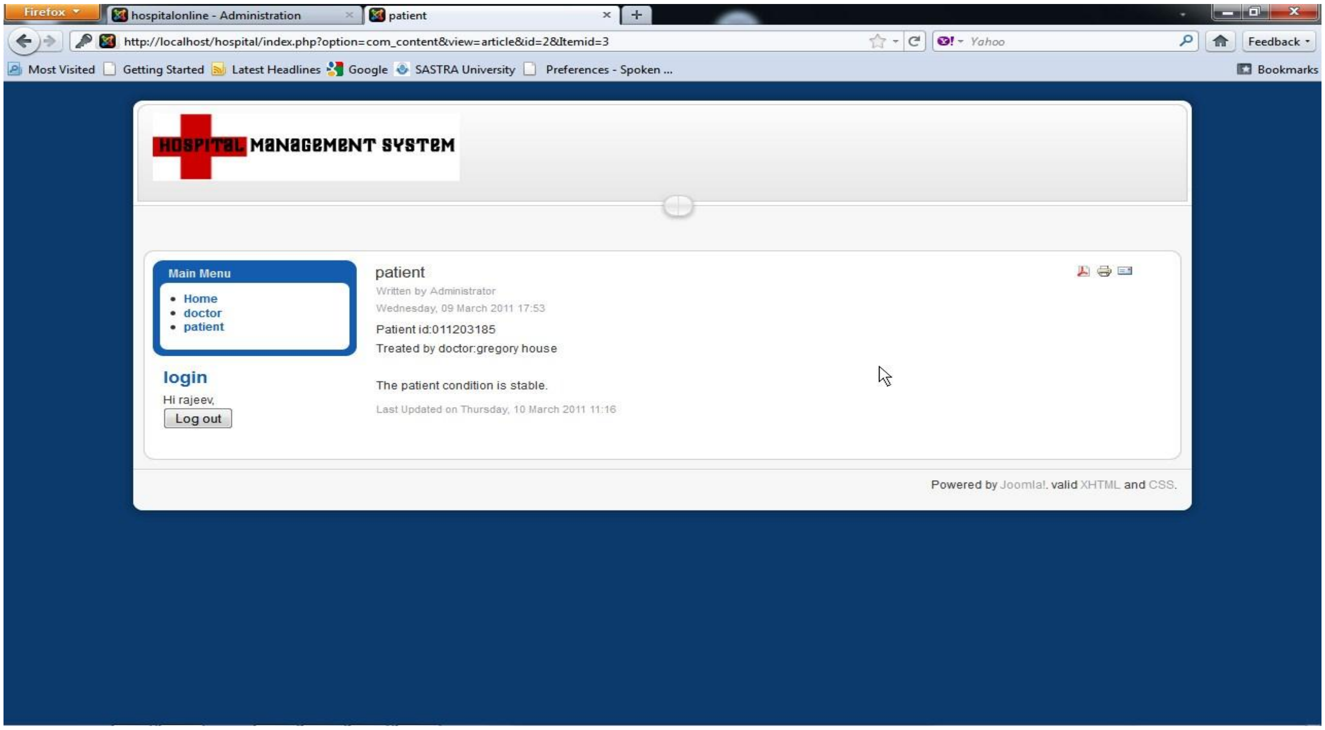

Fig 3.8: Server Database for Caretaker.

4. CONCLUSIONS & FUTURE SCOPE

Thus the above process helps the doctor in so many ways. Primarily the response time of the doctor is reduced drastically. Say there is an emergency situation in an ICU ward. The normal response time (without AAL) would be about 10 minutes in an average general hospital and that would increase for a bigger hospital. But it would take less than 2 seconds for emergency situation to e sent directly to the doctor and slightly more time for updating in the web service if this method is implemented. This even provides a better way of maintaining the patient record since each patient is given account and when he is admitted, his conditions are updated at real time. Since the patient's caretaker can also view the patients account, he can know about the patient's status in real time as the doctor updates the patient's status and diagnosis. Thus implementing this paper will improve the efficiency of the ICU scenario in a hospital.

Future enhancements will include a real time monitoring of the data that is being store in the database.. In case of any discrepancy in the patient's condition, the zigbee module attached to the equipment which triggered the emergency condition sends the emergency condition directly to the doctor who has a zigbee enabled device.

5. ACKNOWLEDGMENTS

The author wants to thank Rengarajan Amirtharajan Assistant Professor / ECE School of Electrical & Electronics Engineering SASTRA University for his valuable guidance and moral support.

6. REFERENCES

[1] Allen Y. Yang , Roozbeh Jafari, S. Shankar Sastry and Ruzena Bajcsy "Distributed recognition of human actions using wearable motion sensor networks ",Journal of Ambient Intelligence and Smart Environments 1 (2009) 103–115

[2] Andreas Lorenz , Reinhard Oppermann" Mobile health monitoring for the elderly: Designing for diversity" Pervasive and Mobile Computing 5 (2009) 478 495

[3] Constantin Timm, Jens Schmutzler, Peter Marwedel, Christian Wietfeld," Dynamic Web Service Orchestration applied to the Device Profile for Web Services in Hierarchical Networks", COMSWARE (2009) Article No: 18 ACM 978-1-60558-353-2 doi: 10.1145/1621890.1621914

[4] Jaime Martín, Ralf Seepold, Natividad Martínez Madrid, Juan Antonio Álvarez, Alejandro Fernández-Montes and Juan Antonio Ortega, N. Martínez Madrid, R.E.D. Seepold (Eds.) "A Home E-Health System for Dependent People Based on OSGI", Intelligent Technical Systems, Lecture Notes in Electrical Engineering (38) (2009) 117-130

[5] Juan P. Lázaro, Alvaro Fides, Ana Navarro, and Sergio Guillén, "Ambient Assisted Nutritional Advisor for Elderly People living at home" Conf Proc IEEE Eng Med Biol Soc. (2010)198-203

[6] Maged N. Kamel Boulos" An Enhanced Ambient Assisted Living Experiment for Older People with Multiple Chronic Conditions" 2nd International Symposium on Applied Sciences in Biomedical and Communication Technologies, Bratislava, Slovak Republic(2009) 24-27

[7] Moreno, P.A. Hernando, M.E. Gomez, E.J. Del Egido, R.P. Matilla, O.B. de Poorter, A. Ortiz, A.H. Vidal, F.G.,

"Teleconsulting: A medical application based on IP multimedia subsystem technology for ambient assisted living" , 2nd International Symposium on Applied Sciences in Biomedical and Communication Technologies (2009) 1-6

[8] V.Moumtzi,, C.Wills, A. Koumpis " Service composition to support ambient assisted living solutions for the elderly" Xii Mediterranean Conference On Medical And Biological Engineering And Computing Ifmbe Proceedings 29(4)(2010) 780–783,

[9] Narayanan C. Krishnan ∗ , Colin Juillard, Dirk Colbry and Sethuraman Panchanathan," Recognition of hand movements using wearable accelerometers", Journal of Ambient Intelligence and Smart Environments, (2009) 143–155.

[10] Martinez-Julia, Pedro Antonio Marin Cerezuela Antonio Marí n Cerezuela Gomez-Skarmeta,Antonio F, "A service oriented architecture for basic autonomic network management" IEEE Symposium on Computers and Communications (2010) doi : 10.1109/ISCC.2010.5546788

[11] Pruter, S., Moritz, G. Zeeb, E., Salomon, R., Timmermann, D., Golatowski, F" Applicability of Web Service Technologies to Reach Real Time Capabilities" 11th IEEE International Symposium on Object Oriented Real-Time Distributed Computing (ISORC), (2008) 229 – 23.

[12] Ratko Magjarevic "Home Care Technologies for Ambient Assisted Living", 11th Mediterranean Conference on Medical and Biomedical Engineering and Computing 2007 IFMBE Proceedings, 16(11) (2007) 397–400,

[13] Ruihua Zhang, Dongfeng Yuan, Yamin Wang, A Health Monitoring System for Wireless Sensor Networks"2nd IEEE Conference Industrial Electronics and Applications(2007) 1648 – 1652

[14] C. N. Scanaill, S. Carew, P. Barralon, N. Noury, D. Lyons & G. M. Lyons, "A review of approaches to mobility telemonitoring of the elderly in their living environment", Annals of biomed. Eng. 34(4), (2006) 547—56.