Development of a 3G Authentication Based Mobile Access of Health Records: A Mobile Telemedicine Application

By

CHUA SENG TEONG

Dissertation submitted in partial fulfillment of
the requirements for the
Bachelor of Engineering (Hons)
(ELECTRICAL AND ELECTRONICS ENGINEERING)

JUNE 2004

Universiti Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

t
RA
445
.T253
2004

1. Telemedicine -- Organization & administr
2. Health Services Accessibility
3. EEE -- Thesis

CERTIFICATION OF APPROVAL

Development of a 3G Authentication Based Mobile Access of Health Records: A Mobile Telemedicine Application

By

CHUA SENG TEONG

A project dissertation submitted to the
Electrical and Electronic Engineering Programme
Universiti Teknologi PETRONAS
in partial fulfillment of the requirements for the
Bachelor of Engineering (Hons)
(ELECTRICAL AND ELECTRONICS ENGINEERING)

Approved by,

(ASSOCIATE PROFESSOR DR. VARUN JEOTI)

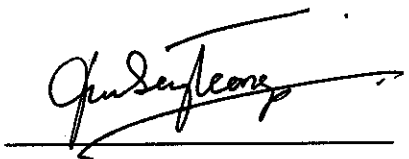
UNIVERSITI TEKNOLOGI PETRONAS

TRONOH, PERAK

JUNE 2004

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein has not been undertaken or done by unspecified sources or persons.



CHUA SENG TEONG

ABSTRACT

As our country progresses in its aim to be a developed country by the Year 2020, the field of Information and Communications Technology or ICT is fast becoming the forerunner for the vision. The Internet is used in almost all aspects of life. As for the communications sector, according to Global Mobile Subscriber Database December 2002 report, there are 8,814,700 mobile subscribers in Malaysia with an Annual Growth of 16.6%. With the adoption of 3G-communication technology in the coming years, compelling high speed services, reaching up to 2 Mb/s together with improved security features would soon be possible.

Through these years in the mobile industry, the health sector has always been neglected. Reason being, the technology could not support the application and it is not so much of a revenue generating business compared to mobile games or sports news. With globalization where the society is always on the move across borders, together with degrading environment conditions and the need for time, instant health services are becoming crucial. Looking into these conditions of mobile adoption and health status, the author intends to develop a solution for a mobile telemedicine application. Kevin Hung (2003) defines telemedicine as the utilization of telecommunication technology for medical diagnosis, treatment and patient care.

Thus, the main aim of this project was to develop an application that could be used for medical purposes. This project integrates the latest mobile telecommunication technologies together with medical services with the idea of providing a highly secured personalize medical system and database query as mobile handsets are becoming a necessity to individuals. This would make updating and retrieving medical health records hassle free, anytime and anywhere. This project has also laid the groundwork for future expansion by incorporating the basic audio and video streaming features.

This report accounts for all the concepts, design works and results of the mobile telemedicine application that has been developed successfully.

ACKNOWLEDGEMENT

The author would like to take this opportunity to express his heartfelt gratitude to the following persons, who have contributed to this project through their help, suggestions, and constructive criticisms.

First of all, the author would like to thank his parents, Chua Chee Huat and Jocelyn Kee, for their continuous support and encouragement. The writer hopes that he can eventually make up for all of the hours he spent away from his family while he worked on this final year design project.

The author is greatly indebted to his Final Year Project supervisor, Associate Professor Dr. Varun Jeoti, who continuously monitored his progress while guiding and advising him for the whole duration of this project. Thank you for being such an inspiring mentor.

Appreciation is also extended to Mr. Nicholas Foo of Nokia Singapore and Mr. Teemu Auvinen of Nokia Finland for providing advice, documents, simulators and mobile handsets for this project development.

Last but not least, the author would like to thank Dr. Chua Chee Peng, Ms. Ang Ai Leen and everyone else who provided useful comments, inspiration and support to help the author complete his project successfully.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF APPENDICES

ACRONYMS AND ABBREVIATIONS

3G - 3rd Generation Mobile System known as UMTS (Universal Mobile Telecommunications System)

3GPP - 3rd Generation Partnership Project

AMR - Adaptive Multi-Rate

BP - Blood Pressure

CGI - Common Gateway Interface

CSD - Circuit Switched Data

CSS - Cascading Style Sheets

DICOM - Digital Imaging and Communications in Medicine

ECG - Electrocardiogram

ETSI - European Telecommunications Standards Institute

GPRS - General Packet Radio Service

GSM - Global System for Mobile Communications

ICT - Information and Communication Technology

IEEE - Institute of Electrical and Electronics Engineer

IP - Internet Protocol

J2EE - Java 2 Enterprise Edition

J2ME - Java 2 Micro Edition

JDBC - Java Database Connectivity

JSSE - Java Secure Socket Extensions

LBS - Location Based Services

LHR - Lifetime Health Plan

MMS - Multimedia Messaging Service

MP - Mobile Profile

ODBC – Open Database Connectivity

OMA - Open Mobile Alliance

PC - Personal Computer

RSA - Rivest-Shamir-Adleman

SDK - Simulation Development Kit

SMIL - Synchronized Multimedia Integration Language

SMS - Short Message Service

SQL - Structured Query Language

SSL - Secured Socket Layer

TCP/IP - Transmission Control Protocol over Internet Protocol

TLS - Transport Layer Security

UDP - User Datagram Protocol

UMTS - Universal Mobile Telecommunications System

W3C - World-Wide Web Consortium

WAP - Wireless Application Protocol

WML - Wireless Markup Language

WTLS - Wireless Transport Layer Security

XHTML - eXtensible Hyper Text Markup Language

XML - eXtensible Markup Language

XSLT - eXtensible Stylesheet Language Transformation

CHAPTER 1

INTRODUCTION

In this chapter, the author introduces the background of the project, the problem statement, objectives and the scope of study. A brief review of what to expect in the coming chapters is also stated.

1.1 Background of Study

Under the 3rd Generation Partnership Project (3GPP), which is an organization joined by the leading mobile handsets manufacturers, there would be a form of standardization in the usage of mobile technologies.

Wireless Application Protocol (WAP) marks the beginning of an era where we could make use of mobile-handsets for browsing purposes. Evolving from WAP 1.0, WAP 2.0 has adopted Extensible Hyper Text Markup Language (XHTML) Basic as the base for its mark-up language. XHTML, developed by the World-Wide Web Consortium (W3C), is the language that will be used to create all content, regardless of whether it is intended for the fixed Internet or the mobile phone world. By narrowing the gap between wired and wireless content, XHTML greatly accelerates the pace at which services can be created and improves the usability of wireless services for consumers.

Other Internet standards that have been adopted in WAP 2.0 include Cascading Style Sheets (CSS), Transport Layer Security (TLS), Hyper Text Markup Language (HTTP) and Transmission Control Protocol (TCP). By specifying how these standards are best used in a wireless environment a better user-experience is achieved. The richer content and multimedia services that will be available in 2.5G-

3G networks are going to be based on these and similar standards and will therefore integrate seamlessly with WAP technology.

The release of WAP 2.0 includes the first release of Multimedia Messaging Services (MMS), a service developed jointly with 3GPP, which allows users to send multimedia messages, combining sounds with images and text, to each other in a fashion similar to sending Short Messaging Service (SMS). Additionally, WAP 2.0 further evolves to WAP Push, which can be used for services such as online auctions, where it is important for users to receive information at the point of interest (i.e., the moment something interesting happens), rather than being forced to actively look for the information.

Due to the fact that WAP 2.0 is an open and interoperable standard, it will be a very valuable component in any future mobile service offering. Ericsson, Nokia and Motorola co-founded the WAP Forum together with Unwired Planet (now Openwave) in 1997 and the forum has since grown to more than 450 members, representing manufacturers, carriers and content developers from all parts of the world. The primary goal of the WAP Forum is to bring together companies from all segments of the wireless industry value chain to ensure product interoperability and growth of the wireless market. Taking for example, a WAP page developed for a Nokia handset using software from Forum Nokia can also be viewed by other handset brands such as Ericsson. This ensures that the product developed is not brand specific.

As for the health sector today, it faces serious and increasing problems in the management of resources for disease prevention, follow-up and remote assistance of patients. The cost of in-patient care is increasingly creating problems for both, patients and social security organizations. In contrast, the market for para-health services and applications, such as physical state monitoring during sports training and the use of health call centers, is becoming increasingly common and available to every citizen. Finally, citizen mobility is increasing, with thousands of citizens crossing country borders daily for purposes of entertainment, leisure, shopping and business. Thus, the introduction of new mobile health and para-health personalized

services, based on 2.5 and 3G technologies will provide new markets and opportunities allowing both, citizens and the industry to profit.

The use of General Packet Radio Service (GPRS) and Universal Mobile Telecommunications System (UMTS) or 3G as the backbone in communication technology is essential due to the need to support a continuous connection to the healthcare center, the high bandwidth required for the transmission of the data (which can easily reach the level of 100 Kbps), the communication costs involved (in GPRS and UMTS, the cost is calculated per Kb, instead of per minute of connection) and the high quality of service required for all health related applications. These are requirements, which cannot be met with current Global System for Mobile Communications (GSM) technology as GSM was more suitable for voice data.

With 3G, images, music, video and other data types would provide the user with richer experience. Additionally, 3G service network seamlessly integrates Internet protocol transport into a mobile service environment with a variety of access networks, opening up many new opportunities for IP-based mobile applications. 3G will also provide access to support services such as authentication, security, and billing mechanisms as well as mobile-specific services such as mobility management and location-based services.

Today the mobile terminal is already a lifestyle accessory but tomorrow, it would be the only trusted device we would use 24 hours a day, 7 days a week.

1.2 Problem Statement

The current problem in society is that there is no user friendly, fast and efficient medical service. The society also faces increasing problems in the management of resources for disease prevention, follow-up and remote assistance of patients. In this fast- pace society, vital medical information retrieval and telemedicine service are very much needed anytime and anywhere.

Under the government initiated Telehealth project, the government is looking into the possibility of using modern technologies for telemedicine in order to promote Malaysia as a regional center for health services. Thus, this project would be inline with the aims of the Teleheath project.

Due to the evolution of the Information and Communication Technology (ICT), mobile handsets have become an essential part of the society where almost every citizen in the country owns a mobile handset. Thus this project would look into the possibility of exploiting the situation of the mass mobile handset ownership that covers all levels of society. By developing a telemedicine service that is compatible with mobile handsets, the public would be able to gain hassle free access to these services simply by using their handset. Furthermore, these services can be accessed at anytime and anywhere, provided that mobile coverage exists.

1.3 Objectives and Scope of Study

1.3.1 Objective

1. To create an integrated system consisting of various individual health care services that is compatible with the medical record and system used in Malaysian health institution and hospitals.
2. To obtain a user friendly solution that can be used on mobile handsets and is compatible with computers as well.
3. To look into possibilities of using mobile-handsets as tools in the medical sector using various telecommunications technologies such as GPRS, picture messaging, video and audio streaming.
4. To design a solution that is able to provide adequate remote assistance.
5. To design a solution that would be flexible and reliable that can be used at anytime and anywhere using mobile handsets.
6. To design a solution that would benefit the society in the near future.
7. To design a solution that can be further enhanced and expanded in the near future.

This project is significant as it deals with potential medical usage that can be ventured into and may be of benefit to society. Furthermore, this project is a workable and marketable project whereby if successful, the author could look into the possibility of introducing this system to the Health Ministry of Malaysia.

1.3.2 Scope of Study

The scope of this project is to create an integrated telemedicine software package that could be used in a majority of new generation mobile handsets.

The author hopes to realize a solution that

1. Could be used at anytime and anywhere,
2. Could be accessed on the mobile phone,
3. Has benefits that outweigh the cost,
4. Is practical and minimal maintenance,
5. Is user friendly.

Among the services that are offered in this project are as follows:

i) Medical examination results,
ii) Health and medical reports,
iii) Added security features using SSL/TLS,
iv) Audio/video streaming using 3G capabilities,
v) Location maps and information of hospitals,
vi) Heart, blood pressure, glucose and temperature data.

1.4 Organization of Dissertation

This section describes the organization of the following chapters.

Chapter 2 lays the foundation of this project. It describes in detail the literature review that was done prior to the commencement of this project. This provides the background information on the research and identifies what others have said and/or discovered. This chapter also includes a brief theory of the various components that were incorporated in this application. This provides the fundamentals that are necessary for understanding the development of the mobile telemedicine application.

The following chapters, Chapter 3 and 4 relate to the development of the application where Chapter 3 describes how the program was developed and includes the methodology, program flow and its architecture. The results that were obtained are then displayed and discussed in Chapter 4.

Finally Chapter 5 concludes the dissertation whereby the author reviews the project that he has completed and the problems encountered throughout the two semesters of work. The author also penned his views on how the application he developed could be further improved in terms of its features, usability, security and quality.

1.5 Chapter Summary

This chapter introduced us to a brief background of the project whereby the problem statement and the scope of study were defined. The following chapter would further describe the background and the foundation where this project is based on.

CHAPTER 2

LITERATURE REVIEW AND THEORY

This chapter reviews literatures as well as papers that have been published under the Institute of Electrical and Electronics Engineer (IEEE) that were used by the author as reference and benchmark for this project. The second part of the chapter reviews the theories behind the implementation of the mobile telemedicine application.

2.1 Literature Review

2.1.1 Health Sector in Malaysia

The objective of health services is to raise and continuously improve the health status of individuals, families and communities. One of the aims for healthcare in the Eighth Malaysia Plan is to have a greater use of IT in the delivery of health services.

According to Multimedia Malaysia: Internet Case Study report; March 2002, in 1998, the Ministry of Health (MOH) had a budget of RM4.2 billion. From the current situation, most government hospitals have access to the Internet, some through dial-up connection and other through a leased line. Four hospitals have a complete ICT system, which means that all medical records are computerized and a profile of each patient is available to doctors. These records can be accessed through the Internet. In the near future, all hospitals would have already been computerized and this is where this project would be looking into ways to exploit this situation.

Telehealth project, one of Multimedia Super Corridor (MSC) flagship initiative was lunched and is officially managed by the MOH for the purpose of spearheading the use of ICT in the health sector. Telehealth consists of four pilot applications which are:

i) The Mass Customised Personalisd Health Information and Education (MCPHIE) – application that disseminates information in the form of databases and educational material.

ii) Continuing Medical Education (CME) - application that seeks to expand the skills of healthcare providers using multimedia and information technologies.

iii) TeleConsultation – application that brings healthcare providers together by using the Internet as a means of communication and information exchange.

iv) Lifetime Health Plan (LHR) - to set up a personalized health plan for each individual, taking into consideration the person's health record.

2.1.2 Implementation of a WAP-Based Telemedicine System for Patient Monitoring

The following is a summary of an IEEE paper entitled *Implementation of a WAP-Based Telemedicine System for Patient Monitoring* by Kevin Hung and Yuan-Ting Zhang that was published in June 2003.

The paper describes the implementation of a WAP-based telemedicine system for patient monitoring that has been developed successfully. It utilizes WAP devices as mobile access terminals for general inquiry and patient-monitoring services. Authorized users are able to browse the patients' general data, monitored blood pressure (BP) and electrocardiogram (ECG) on WAP devices in store-and-forward mode. The applications, written in wireless markup language (WML), WMLScript, and Perl, resides in a content server. A MySQL relational database system was set up to store the BP readings, ECG data, patient records, clinic and hospital information, and doctors' appointments with patients. A wireless ECG subsystem was built for recording ambulatory ECG in an indoor environment and for storing ECG data into the database. For testing, a WAP phone compliant with WAP 1.1 was used at GSM 1800 MHz by circuit-switched data (CSD) to connect to the content server through a WAP gateway. CSD runs on a data rate of 9.6kB/s whereas GPRS is able to reach 171.2kB/s.

According to the paper use of current WAP devices in telemedicine is feasible in areas where the application operates in a store-and-forward, client-server, and low bandwidth fashion. The displayed information is limited to text and low-resolution WBMP static images. When displaying graphical information, it is better to first construct the image at the server, thus reducing the usage of memory and processing time at the device. A WAP-based telemedicine system has been developed in their laboratory and its applications include the following:

- ECG Browsing and heart-rate estimation,
- blood pressure browsing,
- patient record browsing,
- clinic and hospital information inquiry,
- doctor's appointment browsing.

The first three items are designed for doctors and patients' family to monitor patients' status. The other items are inquiry services for patients.

The applications were stored in the content server. The user-interface was written in WML and WMLScript, and executed at the WAP device after it had been downloaded from the server. The other part of the application was written in Perl, and executed within the Linux-based content server providing the common gateway interface (CGI) for more complex tasks. The Perl program can dynamically create WBMP graphics and WML decks upon requests from the WAP device. Graphics displayed were patient photographs, simple graphs, and ECG waveforms. These were first constructed in the server before being sent to the WAP device as WBMP files. All data that the applications accessed and manipulated with were stored in a relational database system. A MySQL database system, consisting of two databases at two different sites, was set up to store BP readings, ECG, patient records, clinic and hospital information, and doctors' appointments with patients. One database resided in the content server and another in a remote PC.

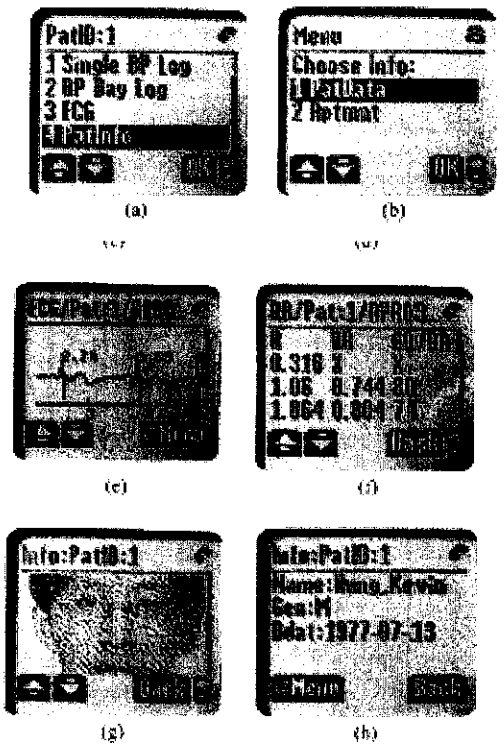


Figure 1: Results obtained from IEEE paper-1

Patient general data, BP, and ECG waveforms were successfully retrieved and displayed on a WAP phone. This is shown in Figure 1 above. Interactive feature extraction of medical data was also demonstrated with performing QRS estimation on ECG data stored in the database. Although response time was long, the feasibility of such a system is expected to improve in the future, as newer versions of the WAP specification will be integrated into the 3G mobile phones, which operate at a much higher data rate and have more on-board resources.

Security is an issue of concern, as in all telemedicine applications. The security features of a WAP-based system are implemented at several levels. WAP implements most of its security in Wireless Transport Layer Security (WTLS) protocol, which is the wireless equivalent of Transport Layer Security (TLS) protocol. The WTLS secure session is only between the phone and the WAP gateway, and not between the device and the content server. The WAP gateway, thus, has access to all of the data in decrypted form. Therefore, using a WAP gateway hosted by a third party is not

recommended for telemedicine applications. The solution is to setup a private WAP gateway for the application.

2.1.3 WAP-Based Personalized Health Care Services

The following is a summary of an IEEE paper entitled *WAP-Based Personalized Heatlth Care Services* by S.Petsas, D.Tzovaras, L.Makris and M.G. Strintzis that was published in year 2001.

The paper presents an approach in the field of mobile access to web-based health-care services where the web-based health-care system not only support web-access but also mobile access through the use of the WAP protocol.

In their three-tier architecture, the Apache Web Server was used. PHP was chosen as the server side scripting language where its usage is similar to that of a Java servlet while MySQL was selected as the Database Server.

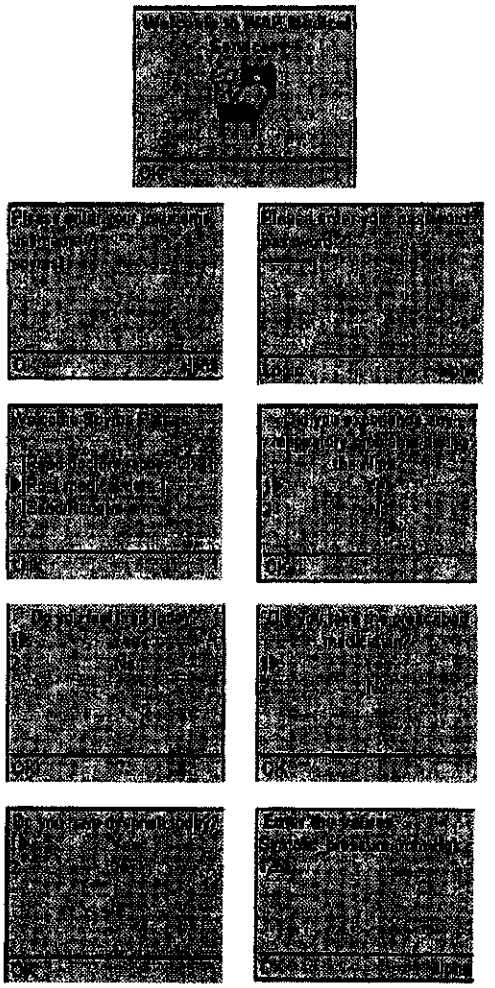


Figure 2: Results obtained from IEEE paper-2

Figure 2 shows the application screenshots that were taken by the proposed system indicating the communication of a patient with the system. Based on the user input via his/her mobile handset, the physician is informed about the condition of his/her patient and also monitors and stores data related to the specific patient.

The issue of security dominates e-health applications and represents new challenges in assuring both patients and physicians that they are operating in a secure environment. Some of the basic elements of security, such as confidentiality, authentication and integrity are addressed by WTLS in this application. WTLS provides encryption and authentication for server-to-client security. This prevents fraudulent access to WAP transactions and opens the way for wireless e-health applications.

2.2 Theory

The following describes the theories behind the various technologies that would be used for the implementation of the project.

2.2 .1 Mobile Browsing

Mobile services are experiencing encouraging growth starting in 2002 and it is expected to grow further. The adoption of packet data (GPRS) and mobile terminals with colour screens has brought benefits to the user, in terms of both, speed and overall experience of mobile browsing. This indicates that next generation mobile browsing technology could form the basis of an attractive business, as browsing remains one of the most important phone applications and services.

With the introduction of XHTML Mobile Profile (XHTML MP) and Wireless Cascading Style Sheets (WCSS) technology, the industry is now able to offer compelling, rich and full colour services to user compared to WAP 1.x which offered black and white browsing and text-based user interface. The effects of WAP evolution is shown graphically in Figure 3 and Figure 4. The specification of WAP 2.0 defined by the Open Mobile Alliance has two key elements, the markup language changes to XHTML Mobile Profile (XHTML MP) and the transmission protocol becomes Wireless Protocol TCP/IP. Both of these standards are standards on the fixed Internet line as well. The advantages of using TCP/IP protocol stack are:

i) Compatible with standard TCP/IP
ii) Faster data transfer for larger files
iii) Better end-to-end security
iv) Able to support more advanced applications

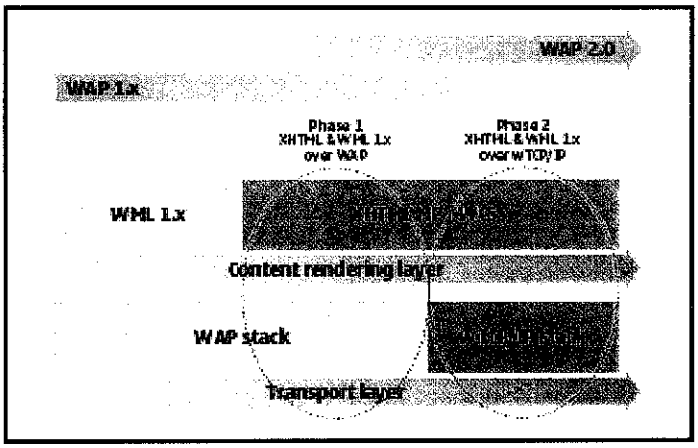


Figure 3: Evolution towards next generation mobile browsing with WAP 2.0

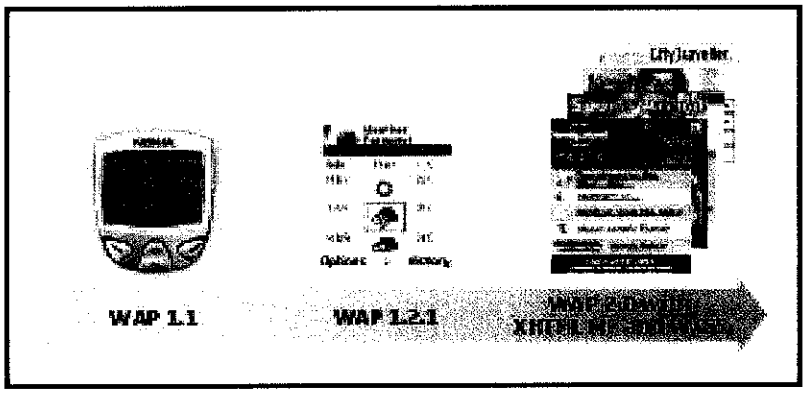


Figure 4: The evolution of mobile browsing – from black and white to rich, full color browsing

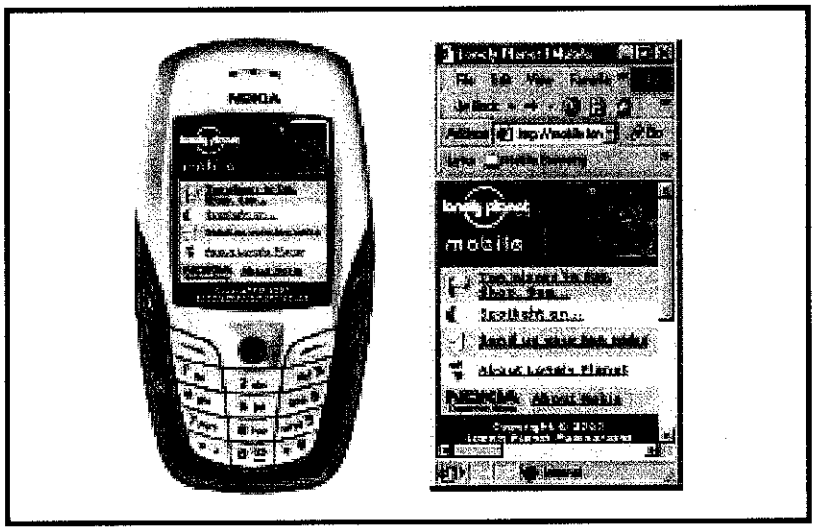


Figure 5: Same XHTML MP document can be viewed both on a mobile device and on a standard desktop browser

(Note: Figures from Nokia White Paper – Next Generation Mobile Browsing ver 1.0)

The idea of using XHTML technology for this project is that the same web-page can be accessed by computers as well as mobile handsets (example is shown in Figure 5). From the second quarter of year 2003, all new mobile handsets launched would support XHTML. The other reasons of using XHTML are:

- Easy, economical and universal, across technologies
- Better user interface and easier-to-use
- Style sheets enable content tailoring to different handsets
- High consumer usability

2.2.2 Java Servlets

Java provides a number of built-in network capabilities that make it easy to develop Internet-based and Web-based applications. Java enables program to search the world for information and to collaborate with programs running on other computers. Java can also enable applets and applications running on the same computer to communicate with one another.

Java servlets allow Java code to be run inside a web server and extends the functionality of the server and to provide similar services offered by CGI scripts. Servlets are now supported by most web servers including Apache, Netscape, Microsoft IIS and others. Using Java servlets instead of CGI programs generally results in more efficient and maintainable code. Java servlets are more efficient because a new process does not have to be initiated for each request. Since Java servlets can be written in simple object orientated language they are often more maintainable than complex Perl scripts. For this application, servlets help provide secure access to a Web site, interact with databases on behalf of a client by incorporating SQL, dynamically generate custom XHTML documents to be displayed by browsers, maintain unique session for each client and most importantly to control the behaviour of the web server.

Java servlets also offer a number of advantages over Java applets. Java applets require the Java code to be run in the user's browser, and suffer from complicated

compatibility problems due to differences between browsers and possibly slow startup due to the need to download many compiled classes into the browser. Java servlets, since they run in the server, will work with any browser and do not result in any code being downloaded to the user's machine.

2.2.3 Video and Audio Streaming

Streaming delivers media from a server over a network to a client in real time. The media is played by the client software as it is delivered. However this is only going to be supported by phones in the near future. As for current phones available in the market, the media is first downloaded into the phone memory.

The standard H.263 currently used was developed to stream video at bandwidths as low as 20K to 24K bit/sec and was based on the H.261 codec, but as a general rule, it requires half the bandwidth to achieve the same video quality [23].

Originally designed as the standard for videoconferencing over ISDN, H.261 introduced features such as motion prediction and block transformation. This allowed for a smoother picture with good quality, but was limited in the amount of motion it could handle. Also, H.261 used a large amount of bandwidth (64K to 2M bytes) and was targeted primarily at circuit-switched networks. The H.263 standard describes only video coding. However, in applications, audio data must also be compressed, transmitted and synchronized with the video signal. Synchronization, multiplexing and protocol issues are covered by "umbrella" standards such as H.320 (ISDN-based videoconferencing), H.324 (POTS-based video telephony) and H.323 (LAN or IP-based videoconferencing). H.263 (or its predecessor, H.261) provides the video coding part of these standards groups. Audio coding is supported by a range of standards including G.723.1. Other, related standards cover functions such as multiplexing (e.g. H.223) and signaling (e.g. H.245) [18].

One of the most advanced speech coding standards today is the Adaptive Multi-Rate (AMR) speech codec, which was developed by the European Telecommunications Standards Institute (ETSI). It includes eight speech coding modes, whose bit rates

range from 4.75 to 12.2 kbit/s. AMR-WB represents state-of-the-art technology in low bit rate wideband speech coding. Like AMR, it is a multi-rate speech codec. AMR-WB technology uses nine bit rates between 6.6 and 23.85 kbit/s at 16 kHz sampling rate. AMR-WB uses file extension *.awb. AMR-WB has been selected by the 3GPP in December 2000 and ITU-T in July 2001. The ITU-T AMR Wideband is now known as G.72.

The coding format is related to the action of a specific coding algorithm that codes the content information into a code stream. The file format is a way of organizing video and audio code streams so that they can be accessed for local decoding and playback or streamed over a transport channel. Currently, only the following two video file formats are supported:

- 3GPP file format (.3gp or .3gpp)
- MP4 file format (.mp4)

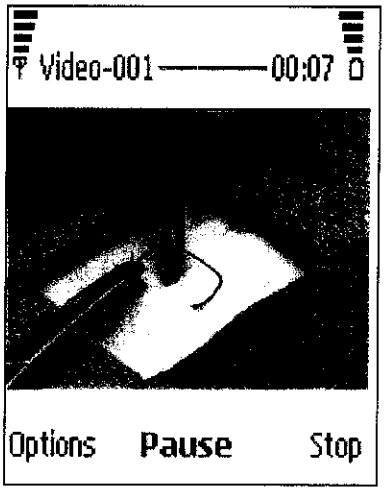


Figure 6: Example of video streaming for medical purpose

2.2.4 SSL / TLS Security

Secure Sockets Layer or SSL, is a technology which allows web browsers and web servers to communicate over a secured channel. It has facilities for protecting data in transit and identifying the user which you are communicating. The secured channel is transparent, which means that it passes the data through unchanged. Thus data being sent is encrypted by one side, transmitted, and then decrypted by the other side prior to any processing [32]. This is a two-way process, meaning that both the server and the browser encrypt all traffic before sending out data.

In WAP versions earlier than WAP 2.0, security is based on Wireless Transport Layer Security (WTLS), which is closely related, but not identical, to SSL, as SSL runs on top of TCP and could not run over UDP or directly over IP. In WAP 2.0, the WAP stack is replaced with the wireless profiled TCP/IP stack, which enables the usage of Transport Layer Security (TLS) which is an improved version of SSL. One prominent feature is its ability to support datagram such as UDP.TLS establish a tunnel in the gateway (Figure 7) and ensures real end-to-end security with no security gap. In other words, there is no security protocol conversion in the WAP gateway [32]. Security is comparable to the Internet model, where transaction all the way from the client to the origin server can be secure. In practice, this means improved security for sensitive applications, such as banking and mobile commerce as well as personal medical records.

TLS is compatible with the security standards set by the Digital Imaging and Communications in Medicine (DICOM) workgroup 14. The DICOM Standards Committee exists to create and maintain international standards for communication of biomedical diagnostic and therapeutic information in disciplines that use digital images and associated data [15].

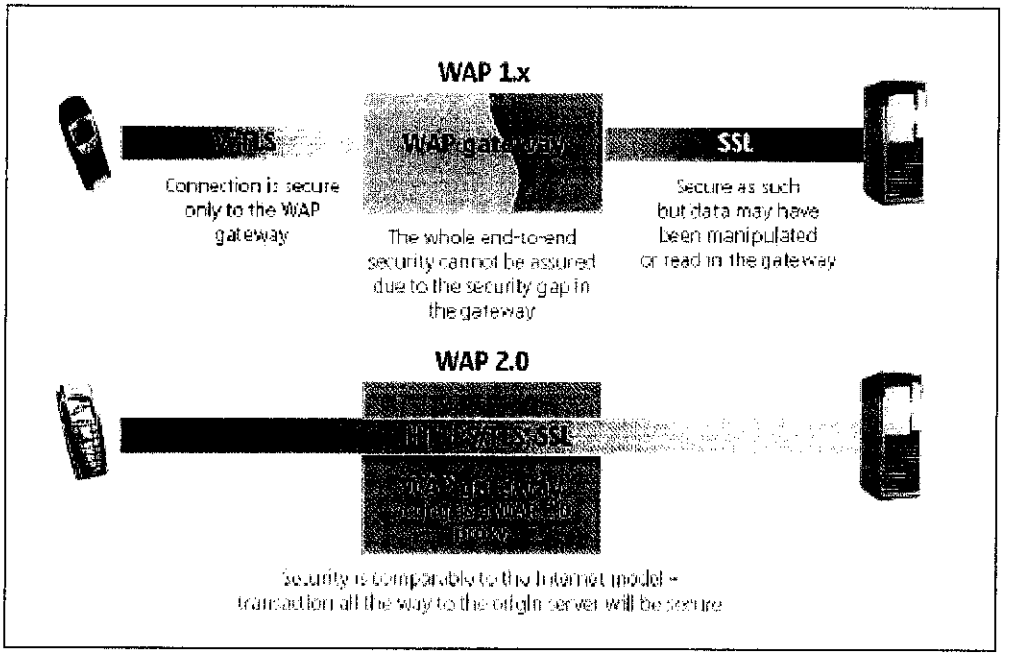


Figure 7: Security Comparison between WAP1.x and WAP2.0

(Note: Picture taken from Nokia_WAP_Phone_security_v3.3_en document)

2.3 Chapter Summary

For the first part of this chapter, the author reviewed the background and foundation of the problem statement of this project. It can be noticed from those literatures reviewed that the projects are quite similar to what the author has developed except that the author improves on the idea by using the latest mobile communication technology. The WML language is replaced by XHTML language which enables higher interactivity and improved graphics.

This chapter also introduces the theory of various technology components such as XHTML browsing, Java servlets, SSL/TLS security which would out phase the WTLS security commonly used currently and video/audio streaming. These concepts are essential to understand the method of implementing the mobile telemedicine application.

The author would be stating the methodology used for this project in the following chapter.

CHAPTER 3

METHODOLOGY

This chapter discusses the methods and procedure used by the author. Among others, the project plan, project flow and architecture and the methods used for implementing each of the project components are explained.

3.1 Plan of the project

The brief project road-map is shown in Figure 8. The author started the project with a basic planning of what services are to be developed for the first phase of the project which was covered in the first of two semesters. The list of services listed under Section 1.3.2 previously was developed and integrated using XHTML, Java servlets and SQL languages.

Under Phase 2 of the project, the author has enhanced the security feature of the project by implementing the SSL/TLS security system. Audio/video streaming capabilities and support for other data types such as graphs and images was also included.

Finally, the author integrates all the components of the project and testing was done to verify its functionality and workability.

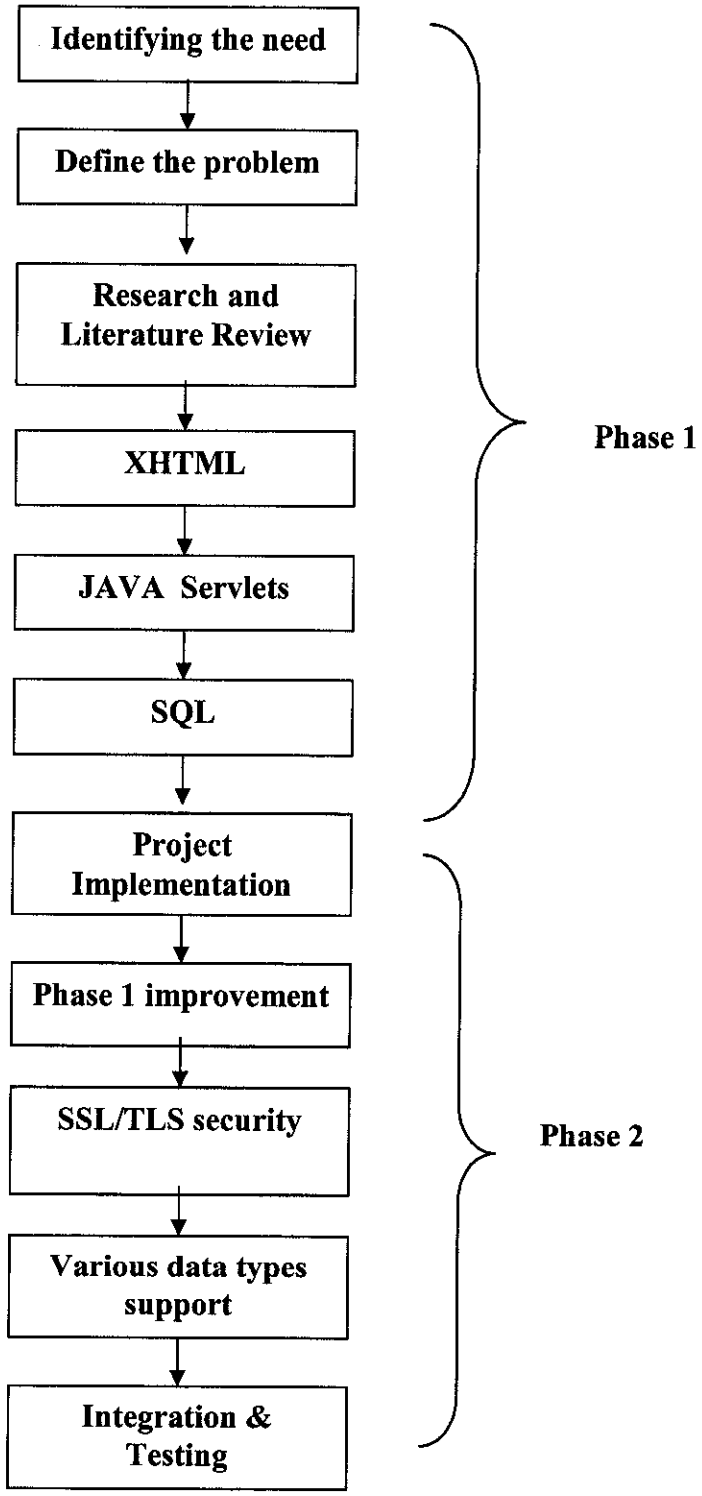


Figure 8: Methodology

3.2 System Architecture

Figure 9 in the following page shows the system architecture of this application. The Apache Tomcat Web Server is the central element in the system where all the server control files, security implementation and servlets reside.

There is a three stage security implementation. There is the SSL/TLS security implementation, which would ensure message confidentiality and integrity. Then the user is challenge with a username and password authentication. Finally, the Apache Web Server would determine the type of user logged in and proceed accordingly.

Should there be any database request the Web Server would query the database server. However for this project, the Microsoft Access database resides in the same machine as the Web Server. For any other data that involves video and audio streaming, the open source Darwin Streaming Server services are invoked by the Tomcat Web Server.

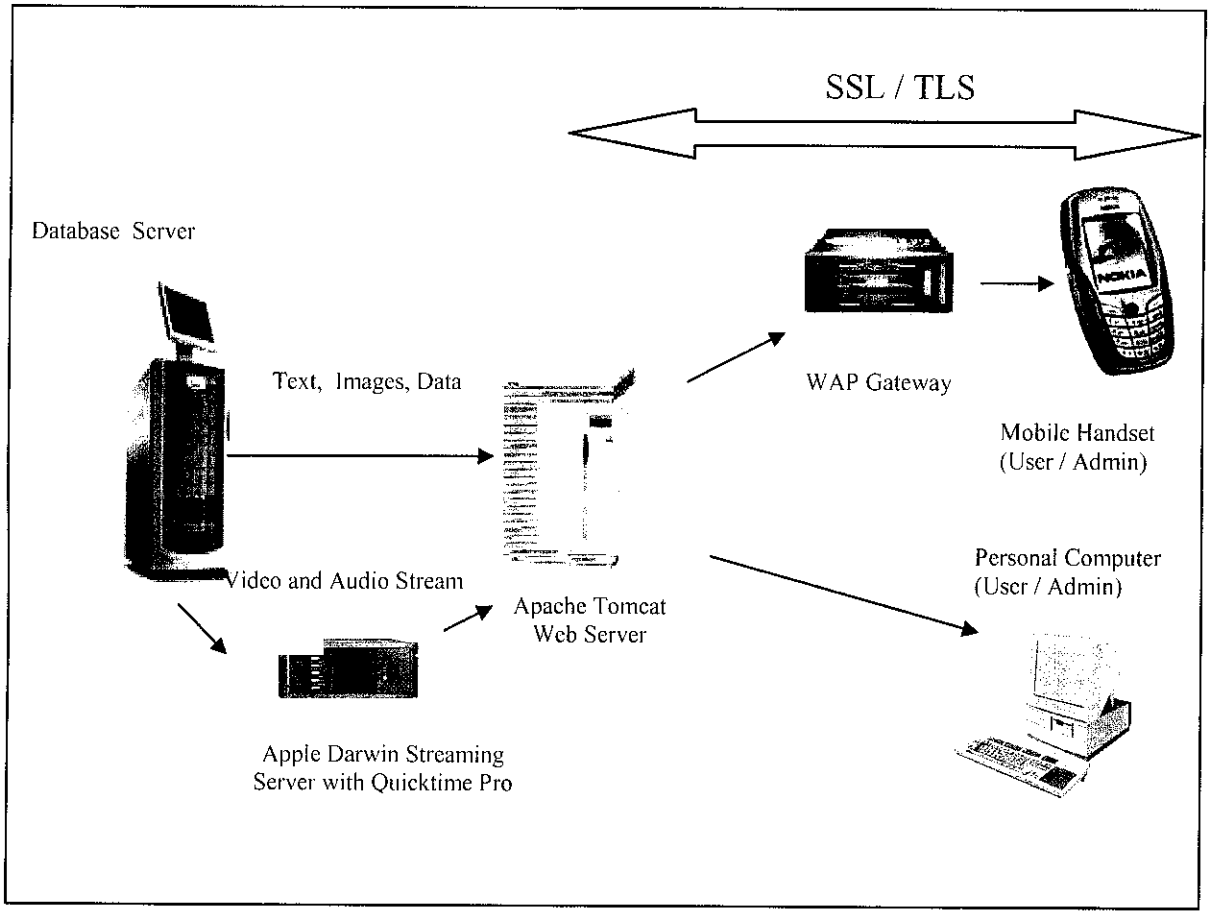


Figure 9: System Architecture

3.3 Program Flow

The flow of the program starts when the user accesses the site via the computer or the mobile handset. Firstly, the web server would try to establish an SSL or TLS negotiation. If it fails, the page would not be able to load. If successful, the log-in servlet would then be process to query the user for his or her log-in name and password. Only if the combination is valid can one proceed.

Once in the main telemedicine page, one can choose various options such as database access or hospital information. However, certain pages such as updating database information are only available strictly to administrators. The user group validation is processed during the login–password validation stage.

The user is able to log-out from the system by accessing a link from the main telemedicine page. For extra security feature, the web server would automatically

disconnect the user if the user is idle for 10 minutes. This is to ensure that the confidentiality of the user is not compromised should the user forgets to log-out. Should the user wish to continue, he or she would have to enter his or her username and password again.

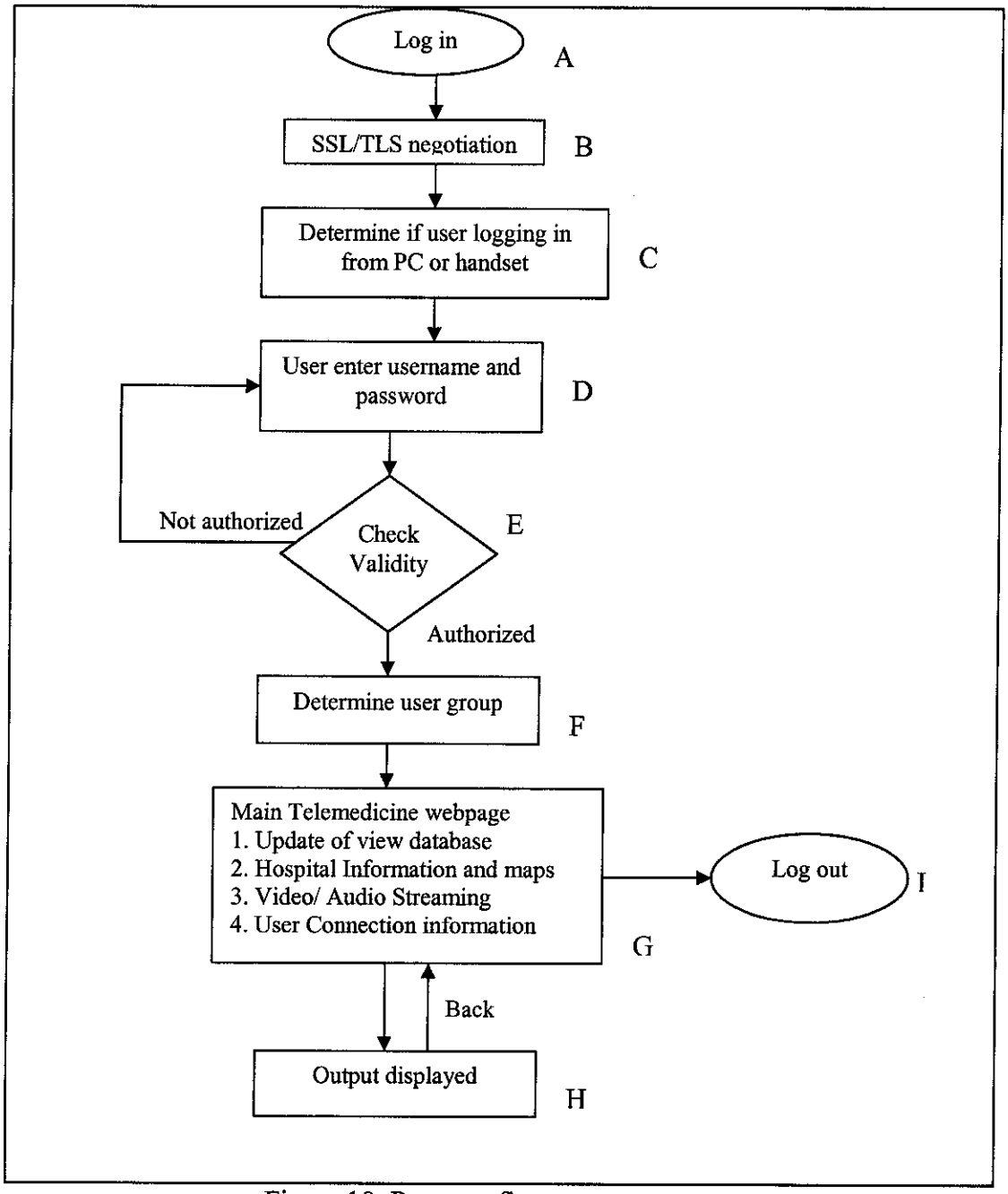


Figure 10: Program flow

3.4 Detailed Breakdown and Implementation of Program

This section describes how each stage shown in Figure 10 was implemented.

For part A, the Web.xml file in the Apache Tomcat Webserver is modified to include an authentication function. The Web.xml document plays an important part as it is the control module of the whole web application. There are various types of authentication such as basic, digest, form or custom authentication. The author implemented a form based authentication and the following codes are included in the Web.xml file

```
<login-config>
    <auth-method>  FORM  </auth-method>
    <form-login-config>
      <form-login-page> /security/loginpage.xhtml
      </form-login-page>
      <form-error-page> /security/errorpage.xhtml
      </form-error-page>
    </form-login-config>
  </login-config>
```

Part B is a crucial part of the project whereby the TLS security system is implemented. Firstly, the following codes have to be included in the Web.xml file :

```
<user-data-constraint>
        <transport-guarantee>CONFIDENTIAL</transport-guarantee>
</user-data-constraint>
```

With the Java Secure Socket Extensions (JSSE) package, the author creates a self-signed certificate using the following command. RSA algorithm is the preferred algorithm as it ensures compatibility between various platforms.

```
C:\j2sdk.1.4.2\bin\keytool -genkey -alias tomcat -keyalg RSA
```

Using the open-source PureTLS program, the author is able to obtain the following files, cryptix32.jar, puretls.jar, and cryptix-asn1.jar which without these files, TLS

cannot be implemented. Finally, the following codes are included in the Server.xml file of the web server to activate TLS security.

```
<Connector port="8443"
      maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
      enableLookups="false" disableUploadTimeout="true"
      acceptCount="100" debug="0" scheme="https" secure="true">
   <Factory clientAuth="false" protocol="TLS" />
</Connector>
```

Part C, D, E and F are closely related. Firstly the server determines if the user is logging in using mobile handset or a computer. The difference between the XHTML source codes for a mobile handset compared to a computer is the heading of the XHTML code. For mobile handset, the XHTML code is as follows :

```
<!DOCTYPE html PUBLIC "-/WAPFORUM//DTD XHTML Mobile 1.0//EN""http://www.wapforum.org/DTD/xhtml-
mobile10.dtd">
```

The XHTML heading to support a computer web browser is:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML xmlns="http://www.w3.org/1999/xhtml">
```

Part D would display the login page where the source code is shown in Appendix D. Then for part E, the server would automatically compare the login page user inputs to that of the database. Finally, the following codes below are included in the Web.xml to implement the user grouping as shown in Part F of the project flow.

```
<security-role>
    <role-name> manager </role-name> <role-name> user </role-name>
 </security-role>
```

Part G which is the main telemedicine webpage is where the user would be redirected to should he or she is an authorized user. The source code of the page is shown in Appendix D.

Part H consists of database access to various types of media such as text, graphs, pictures and audio/video streaming. For text and data, the Web Server connects with the Microsoft Access database through a JDBC-ODBC driver. Microsoft's ODBC which stands for Open Database Connectivity serves as an interface for accessing database. However, it requires the used of Java Database Connectivity to be compatible with the Java platform that the application is running on. The code for the JDBC-ODBC driver is *sun.jdbc.odbc.JdbcOdbcDriver*. For images they can be accessed using through the JDBC-ODBC interface or they can be accessed directly through XTHML. An example using XHTML language would be :

```
<a href="hos_fatimah_edited.gif">
```

For audio/video streaming, a Quicktime plug-in program should be installed in the computer. This plug-in would be automatically called upon should the web browser encounter an audio/video stream file. However, before the file could be streamed, it needs to be converted into the .3gp file format which is the standard set by 3GPP. The author uses Quicktime Pro software to do the file conversion. To access a streamed file using the web browser, the following codes are included:

```
<object CLASSID="clsid:02BF25D5-8C17-4B23-BC80-D3488ABDDC6B" WIDTH="200"HEIGHT="150"
CODEBASE="http://www.apple.com/qtactivex/qtplugin.cab">
<param name="SRC" value="aa002-3G.3gp">
<param name="AUTOPLAY" value="false">
<param name="CONTROLLER" value="true">
<param name="scale" value="tofit">
<embed SRC="aa002-3G.3gp" WIDTH="200" HEIGHT="150" AUTOPLAY="false" CONTROLLER="true"
type="video/quicktime" scale="tofit" PLUGINSPAGE="http://www.apple.com/quicktime/download/">
</embed>
</object>
```

For mobile browsing, the following code would activate the onboard video/audio player :

```
<a href="rtsp://localhost/aa002-3G.3gp">
```

Finally for the graph generation feature, the JDBC-ODBC driver is used to retrieve the data from the database. It is then processed using the open-source KavaChart java extension files and the resulting output would be in a form of a picture file such as .jpeg or .gif where the user is able to save the picture for future reference. The full source code is shown in Appendix C.

Finally, the author implementation of the log-out command is as follows:

```
<% if (request.getParameter("logoff") != null) {
session.invalidate(); response.sendRedirect("loginpage.xhtml");
return; }%>
```

The server would also automatically log-out the user if the user is idle for more than 10 minutes. The following codes are included in the Web.xml file to enable this feature.

```
<session-config>  <session-timeout> 10 </session-timeout>  </session-config>
```

The full source code for the entire implementation of this mobile telemedicine application can be found in the appendices at the end of this dissertation.

CHAPTER 4

RESULTS AND DISCUSSION

The following discusses the results obtained from the mobile telemedicine application that has been developed. The XHTML WAP pages would be simulated using a proprietary Nokia phone simulator provided by Nokia. A web server environment have been set up using the open-source Apache Tomcat Web Server for simulation purpose whereby the author would be using an IP address of 127.0.0.1, which is a local loop back address that points to the author's server for simulation and testing.

The database that was used for this project is the Microsoft Access Database. This is due to the fact that it is one of the more popular database systems being used in the market. Nevertheless, if any other database system is being used such as Oracle Database or MySQL database, the application would require only one line of source code modification to work.

The results are classified under two categories, which are:

i) Security and authentication

ii) Database system which includes various types of data which include text, graphs, images, audio and video

4.1 Security

Due to the TLS security feature, the moment the user tries to log in to the website a warning message would inform the user that he or she is about to enter a secured website and if he or she would like to continue. This is shown in Figure 11.

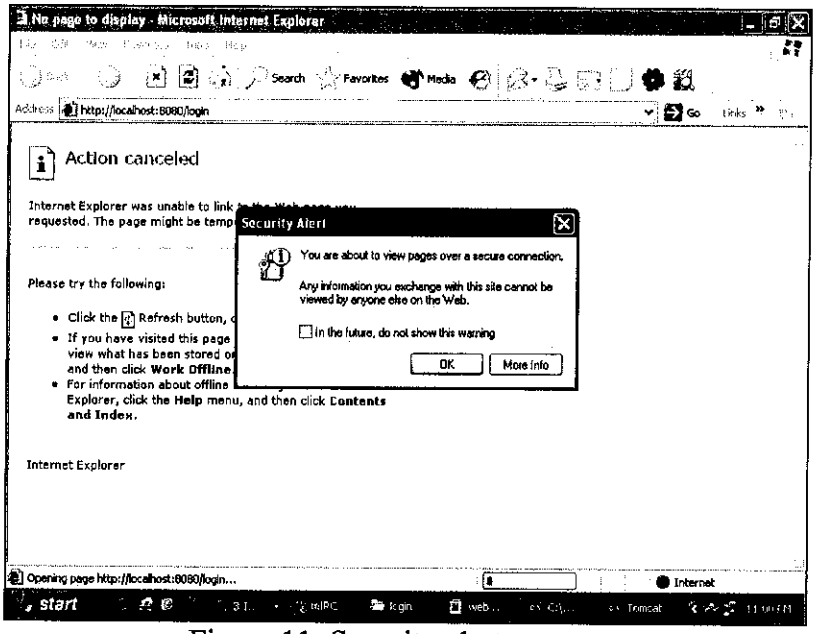


Figure 11: Security alert message

The following screen shot shown in Figure 12 would be displayed to the user if the security certificate was issued by a company or individual that the user have not chosen to trust. Thus the user would have three options. He or she can choose to proceed, to cancel or to view the details of the certificate first. If he or she chooses to view the certificate, the following screen as shown in Figure 13 would be shown.

The certificate shown in Figure 13 is self generated by the author. One is able to view the type of encryption method used for the public key and thumbprint. One is also able to view details such as who are the issuer, validity dates and other vital information.

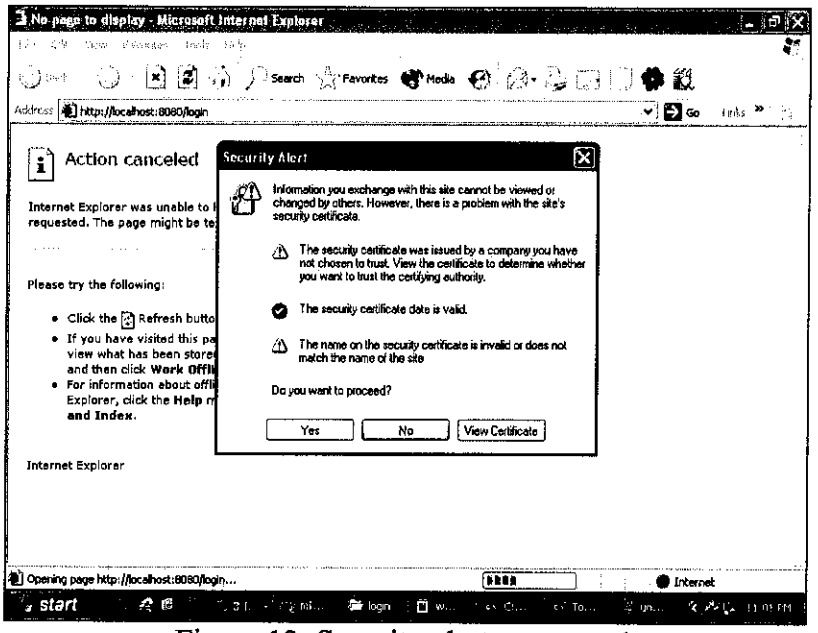


Figure 12: Security alert message-1

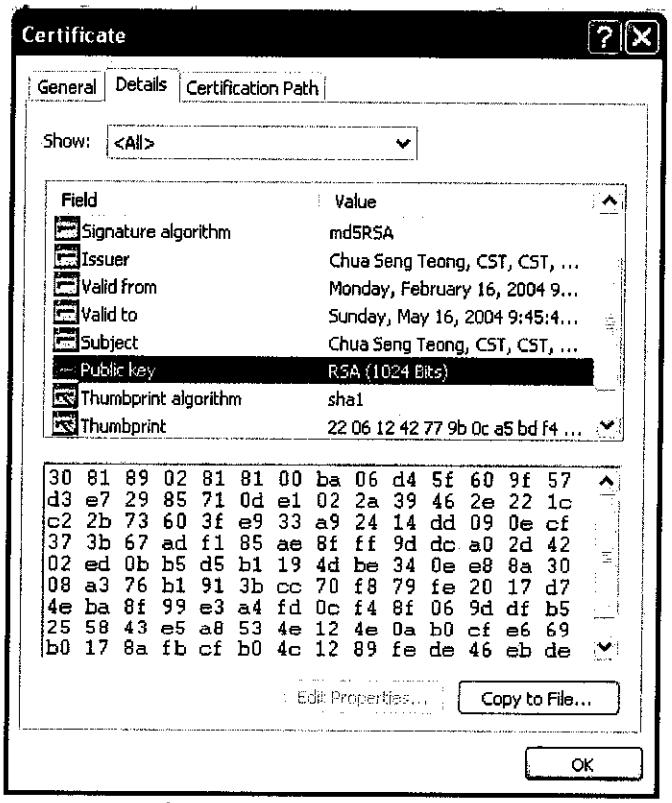


Figure 13: Certificate details

For authentication purposes, the author has chosen form based authentication type. For this form based authentication, the user is challenge with the user name and password the moment the user is logged on to the page. Once the entries are entered

and the "OK" button is selected, the username and password is compared to the authorization list configured in the Apache Tomcat Web Server.

Should the user type the wrong combination of username and password, the following screen shown in Figure 15 would appear. The user then has to click the back button available on the phone to return back to the login page. If the user has type in the correct combination of username and password, the user would be granted permission to access the main page shown in Figure 16.

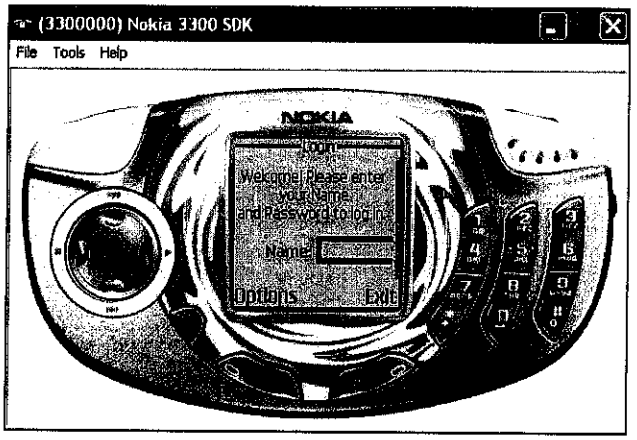


Figure 14: Loginpage.xhtml

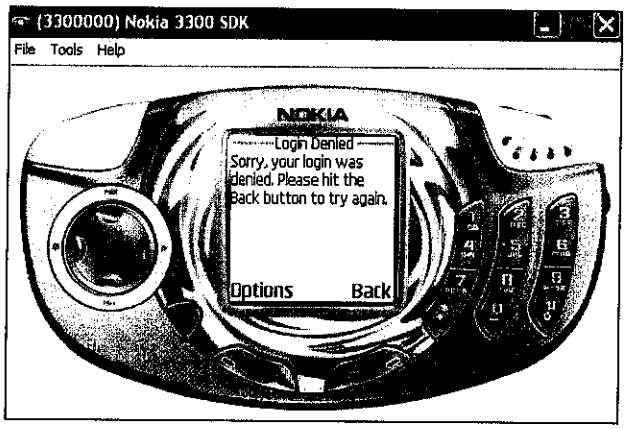


Figure 15: Errorpage.xhtml

During the log-on process, the server would also determine the type of user group the user is in. The author has categorized two user groups which are the user and administrator. This serves as an extra security measure as with this grouping, we are

able to control the users, the operations they perform and able to determine which resources they are allowed to access and which is restricted.

For additional security measure, a session timeout is implemented, in this case 10 minutes. This determines how long the server would wait in between request before invalidating the session. Shorter timeouts increase the user security because it limits the time of vulnerability should a user forget to log out.

4.2 Database

The main page for this WAP-XHTML telemedicine application is shown in Figure 16. This page is where the user is able to select the services included for this mobile telemedicine application. A menu is displayed and the user would be able to select the relevant service by clicking on the links. From this links, medical records which can consist of text, images, graphs and data and even audio and video clips can be accessed. The architecture of this system was shown in Figure 9 from the previous chapter.

The user is able to access this page using the mobile phone and also using the Internet Explorer which is available in any Windows based personal computer. This is where the advantage of XHTML lies in.

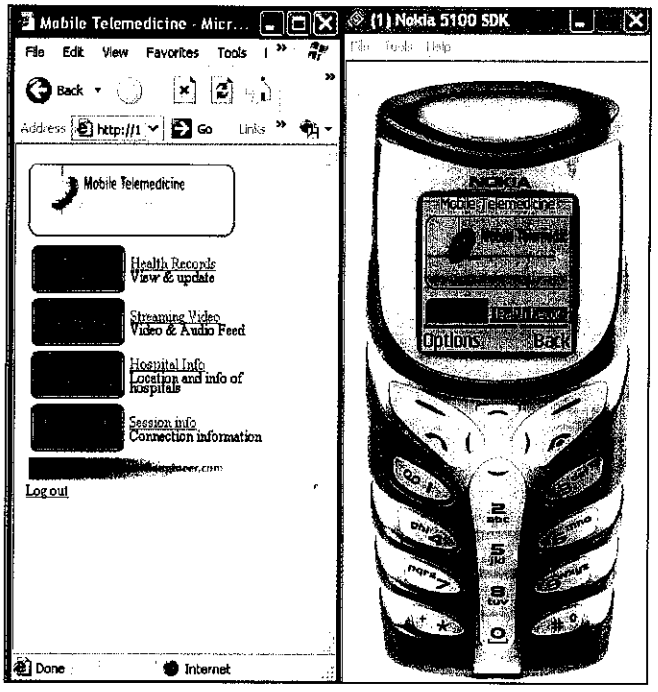


Figure 16: Telemedicine.xhtml

Using the SQL language, we are able to not only query the database but we are able to update or modify a database entry by choosing either the option of viewing or updating the database (shown in Figure 17). This is one of the cases where a need for certain regulations and administrator rights to control the amount of access a client has to the database. This feature would prevent unauthorized modifying of data.

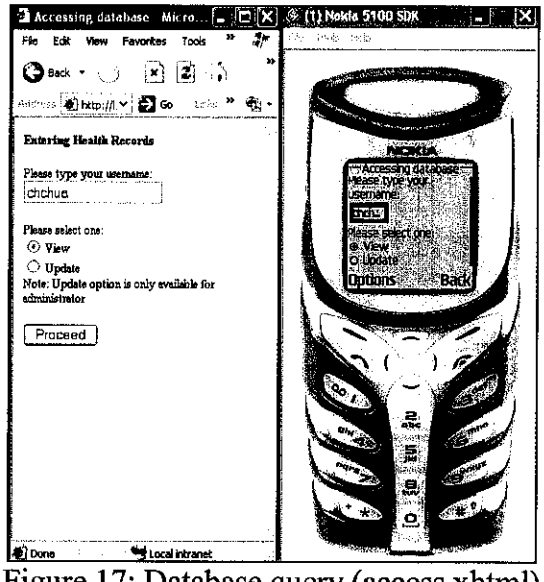


Figure 17: Database query (access.xhtml)

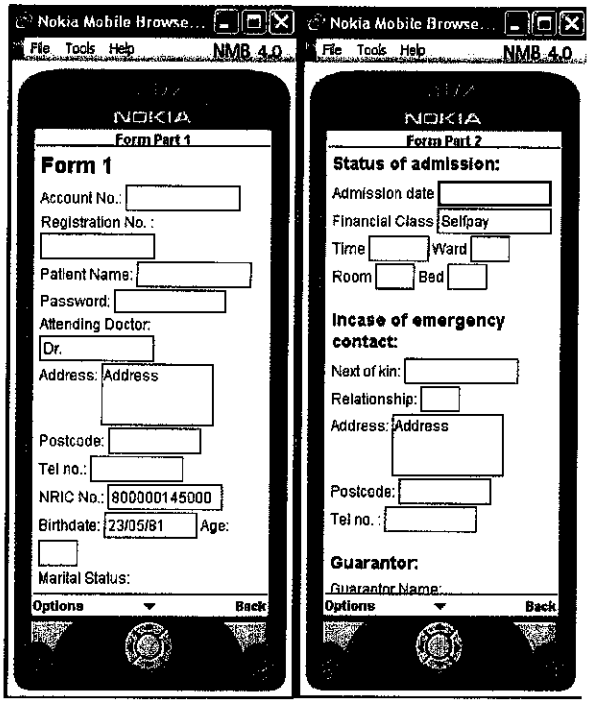


Figure 18: Database update

If the option of updating the database is entered by the user, the resulting screenshot as shown in Figure 18 above would appear whereby one would be able to enter the relevant details of a person's medical records for updating purposes. One would have to press the "Submit" button at the end of the page before proceeding to the next page for the updates to execute.

Figure 19 and 20 shows the results of the database query from the mock medical records created by the author.

As medical records often include information about a patient's blood pressure, temperature, respiration and pulse, this application has been developed to cater for these types of records which are usually shown in graph format. Figure 21 shows the graph that has been generated using data from the database on Internet Explorer and Figure 22 is the equivalent results on the handset simulator.

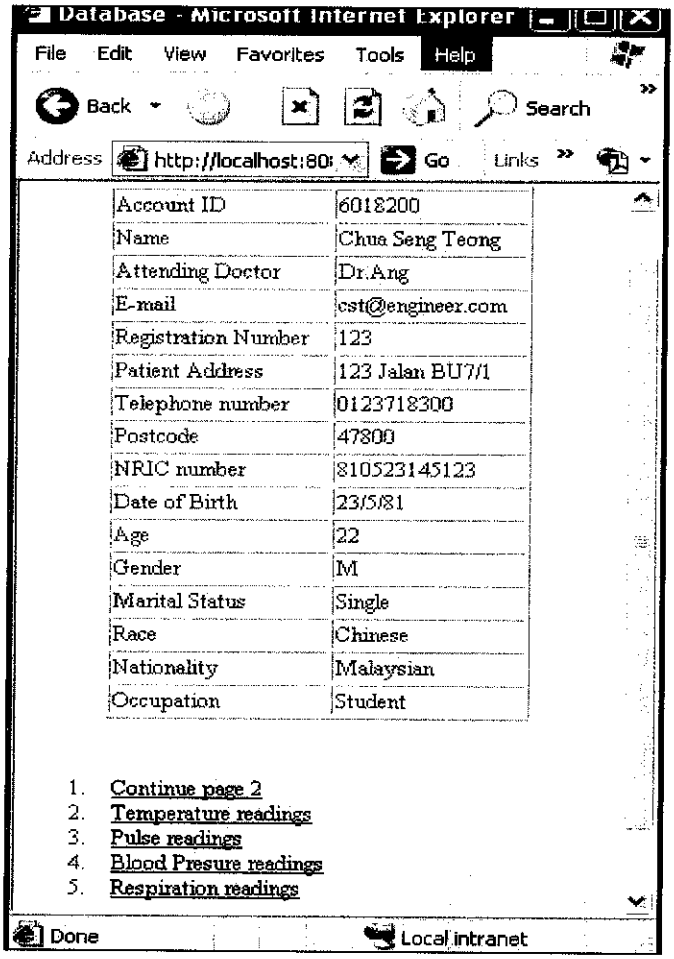


Figure 19: Database query result on Internet Explorer

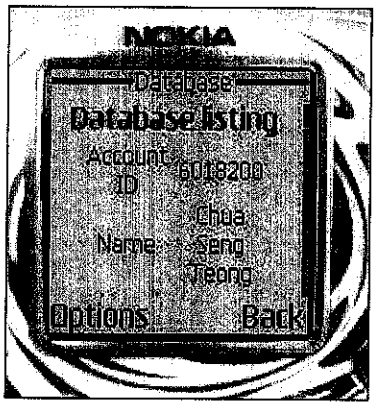


Figure 20: Database query result on Nokia Simulator

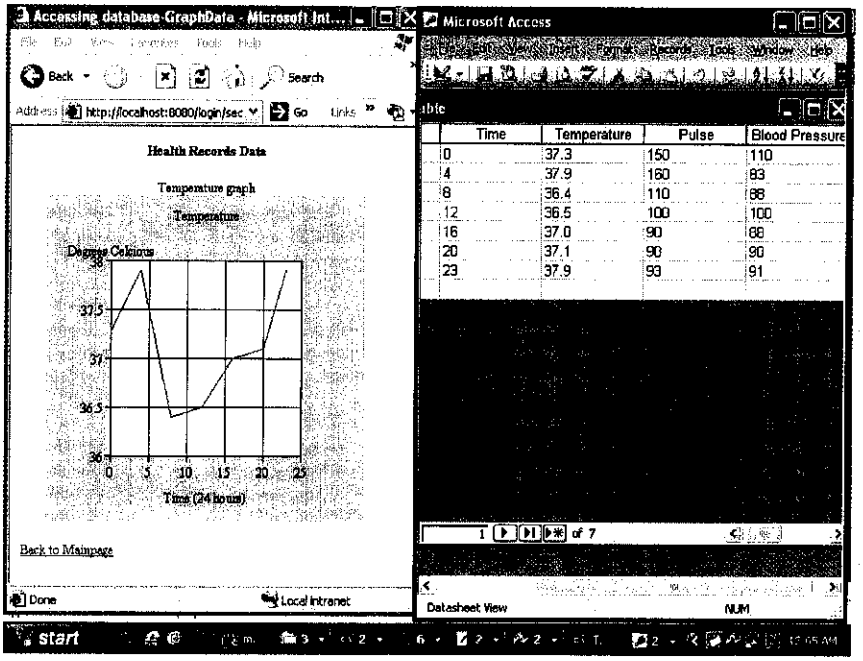


Figure 21: Resulting graph generated from database

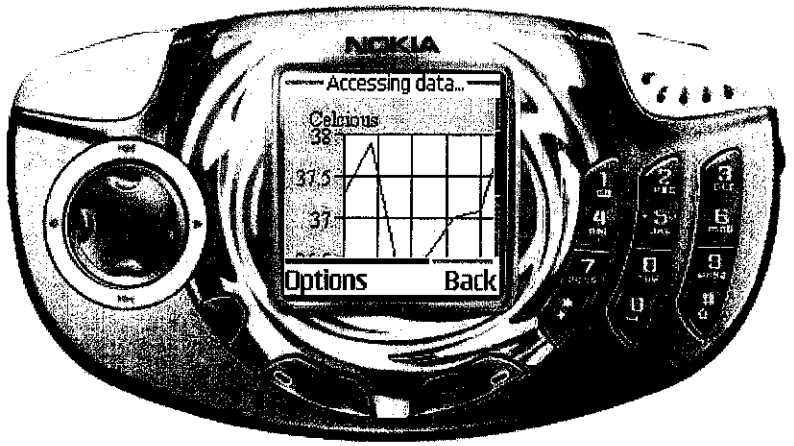


Figure 22: Graph simulated on Nokia Simulator

From this Telemedicine application, one is able to access information about hospitals around the vicinity, which includes a location map of the hospital. This shows the capability of the program to output graphic pictures such as jpeg or gif files. This program can be further improved to be a real-time Location Based Service (LBS) system when the technology advancement permits.

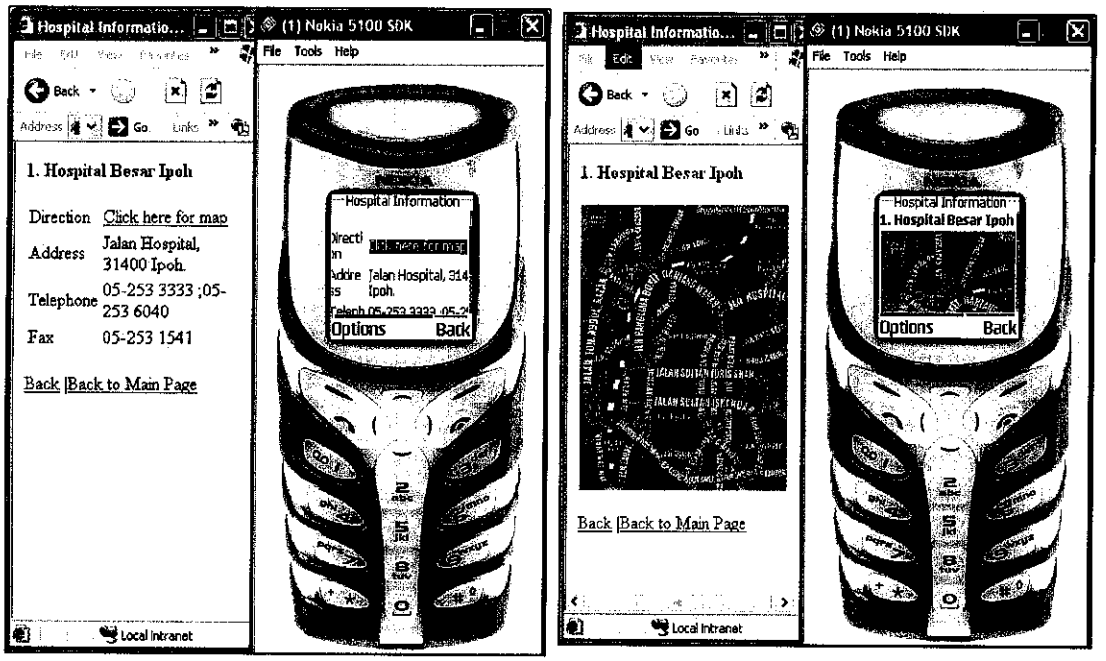


Figure 23: Hospital Information and Location

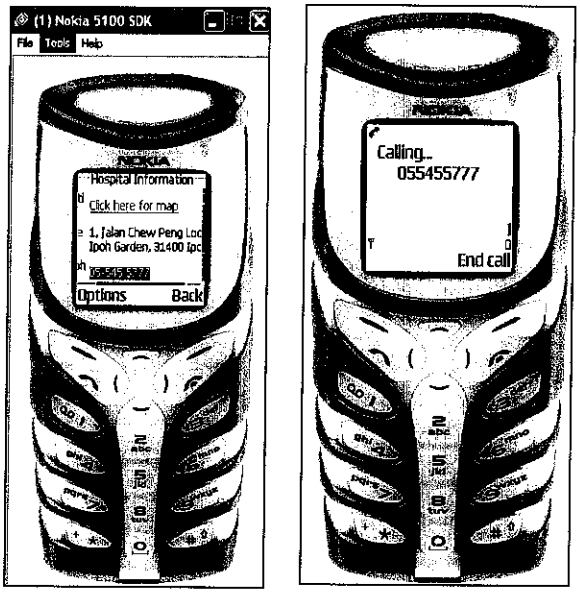


Figure 24: Instant dialing

For emergency purpose where time is crucial, this application supports instant dialing whereby the moment a specific link is entered, the pre-specified number is dialed. This is shown in Figure 24 where the screenshot on the right shows the process of dialing the number specified in the screenshot on the left.

The screen shown in Figure 25 is a screen shot on how a video and audio stream would look like. One has the option to play, pause or stop the video.

Before a video file (any format such as .avi, .mpg..etc) can be streamed, it has to be converted to 3GPP complaint formats as mentioned under Section 2.2.3 previously.

Currently, there is a disadvantage with the mobile phones in the market, which is their limited on board memory, thus the size of the video and audio files that can be streamed is very limited.

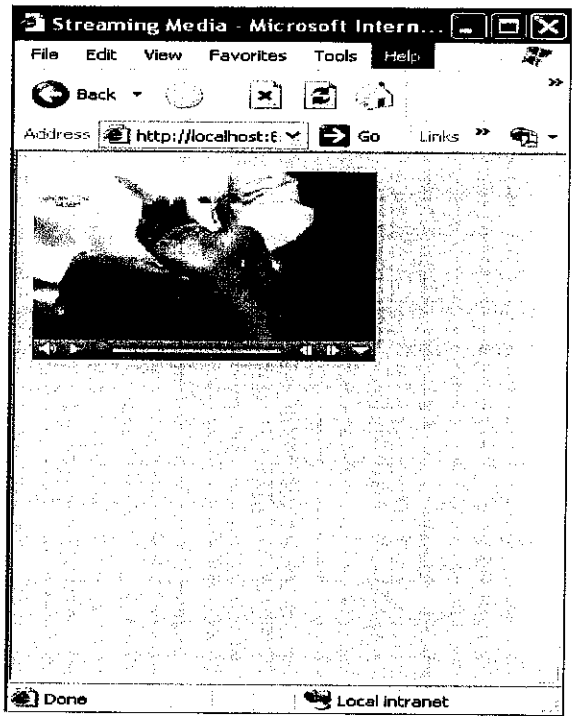


Figure 23: Video/Audio Steaming

4.3 Additional feature

A session information page is included together in this application. Information regarding the log-in such as the number of times the user has accessed the page is included. The administrator can make use of this user information for log and tracking purposes as with the session ID is included.

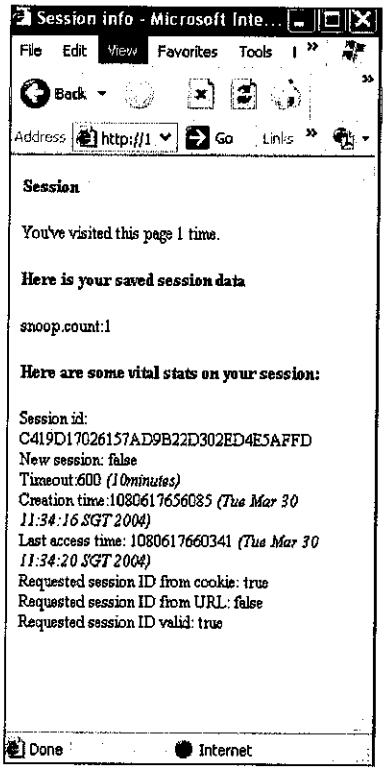


Figure 24: Session information

4.4 Chapter Summary

This chapter illustrated the successful implementation of the application. The TLS security feature, which is compatible with the DICOM standards have been applied together with the telemedicine application. The author has also demonstrated the capability of this application in displaying various forms of data. Results are shown in the form of screenshots taken from the mobile telemedicine application simulated on Microsoft Internet Explorer and Nokia phone simulator.

One is able to see from here that while maintaining its functionality, the project is user friendly and incorporates aesthetic look and feel.

The following Chapter 5 would review and conclude the project.

CHAPTER 5

CONCLUSION AND RECOMMENDATION

This chapter reviews the project, highlighting some of the problems faced and steps taken to overcome the problems. Some recommendations are made to suggest future work for expansion and continuation of the project. The chapter is rounded off with a conclusion.

5.1 Review

This project was developed under two phases, which was carried out over two semesters. The first semester was mainly dedicated to the preliminary research work, learning the XHTML and Java language and developing the main module of the application. The second semester work was focused on security improvements; various database representation and researching various additional add-on modules to be included in this application.

Since the project started, the author has gained various concepts and knowledge for the successful implementation of the final product. Most notable is the XHTML language. The XHTML language is highly structured and the language conforms to stricter rules compared to its predecessor, HTML. Various knowledge regarding mobile handsets, WAP and telecommunication systems was also researched.

Java servlets was another key element that the author has most benefited in. The author is now able to write servlets that processes requests from the XHTML page. Running parallel to the study of Java servlets was a study on Web Server implementation. The author managed to install and configured the Apache Web Server to be able to run the servlets and XHTML pages. The key element to controlling the display and initialization of the servlets in the Web Server is the Web.xml document (shown in Appendix B).

There is a feature in the Java language called JDBC that enables codes written in java language to open, search, select and modify database documents such as Microsoft Access documents using SQL language. The author has successfully implemented this function in the Java servlet.

Eventhough the form authentication has been included; it is by no means secured from hackers and intruders. Thus the author has learned and reviewed higher security standards such as SSL, TLS, WPKI, WTLS and finally implemented the TLS security system. TLS was chosen based on its improved features compared to the other security systems available. Furthermore, it is compatible with the DICOM standard which defines the global standard for transferring medical data.

5.2 Problems Encountered

Two major problems where encountered during the implementation of this project.

Firstly, there is a shortcoming with the Nokia Simulator as it is unable to simulate audio and video streaming. Furthermore, since currently there is no other simulator that is able to show the result of video streaming, thus the author is unable to simulate the audio/video streaming on mobile handset simulators. However, the author has sent the source code for verification of workability to Nokia experts and they have verified those codes.

There was a problem with the integration of the XHTML based system for the telemedicine application that forms the core of the application with Java 2 Micro Edition (J2ME) applications. This is because the Apache Tomcat web server and the Java servlet do not support J2ME files. J2ME files are unlike the normal Java files as there is only limited amount of functions available and they are still relatively new thus many other programs and systems such as the web server have not integrated their system with J2ME support. In fact, another separate server such as a Delivery Server needs to be used to support J2ME. However, due to the problem of integrating the Web Server together with the delivery server, the author decided to developed a full Java servlet and XHTML based application.

5.3 Recommendations for future expansion and continuation

This section presents some recommendations for future expansion and continuation work to improve the design of this mobile telemedicine application.

This project was based on the future generation of WAP browsing which utilizes the XHTML language. This project can also be extended to cover older versions of WAP. With legacy support, this application can cover a broader range of society. However, there would definitely be certain limitations such as the video/audio streaming application.

There were certain ideas based on Location Based Service (LBS) such an emergency signal, individual tracking or directions to the nearest hospital. However, these services cannot be implemented even-though it is an essential service for telemedicine due to limitations of current technology at the time of writing. Telecommunication companies do not support this LBS technology. The closest one could get to LBS is to know the location of the Base Transmission Station (BTS) that is serving the user.

Currently, the graph application that is available is only able to retrieve static data in a patient's database and does not support real-time data presentation. Thus, this graph presentation can be extended to cover real-time data and better interactivity such as locating the exact value of a certain X-Y coordinate on the graph.

The video/audio streaming application developed here opens up a whole new world of possible services that can be offered such as multimedia messaging, health education, or even remote patient consultation. With such services included, it would further increase the functionality of this application.

Finally, for additional security, the application can be extended to have some sort of identification using information from the phone SIM card and maybe together with the mobile handset's PIN number. This would truly verify the identity of the user logging in.

5.4 Conclusion

The mobile telemedicine application has been developed successfully whereby one is able to log-in to the main page and access the health records database which consists of various forms of data. The application was simulated using a Nokia phone simulator as well as Microsoft Internet Explorer and a mock medical database that had been created using the Microsoft Access Database software.

With the successful implementation of this system, the author has demonstrated that an attractive and secured health records database system is possible which includes various text, data, images, audio and video streaming that is packaged together and can be easily access using a mobile handset.

There are still many improvements and additional service that could be added to the current application developed here. It is hoped that this application would one day be widely used throughout the nation and being able to contribute to the society.

REFERENCE

[1] Nokia Mobile Phones "Forum Nokia", www.forum.nokia.com/seap, 2004.

[2] Ericsson "Ericsson Mobility World",http://www.ericsson.com/ mobilityworld/, 2004.

[3] Sun Microsystems, "Java Technology", http://java.sun.com/, 2004.

[4] Open Mobile Alliance Ltd, " OMA Web", http://www.openmobilealliance. org/documents.asp, 2002.

[5] Open Mobile Alliance Ltd, "WAP Forum", www.wapforum.org, 2003.

[6] Apache Software Foundation, "Tomcat 5 servlet",http://jakarta.apache.org/ tomcat/tomcat-5.0-doc/, 2003.

[7] Jason Hunter, "Servlets.com", http://www.servlets.com/doc, 2003.

[8] Karl Moss, "ServletGuru.com", http://www.servletguru.com/downloads.htm, 2002.

[9] O'Reilly Media Inc.,"ONJava.com", http://www.onjava.com/pub/a/ onjava/2001/03/29/tomcat.html, 2001.

[10] University Of Essex, "Jdbcodbc-servlet", http://cswww.essex.ac.uk/ TechnicalGroup/, 2003.

[11] Chris "Sloppy code library", http://www.sloppycode.net/sloppycode/,2004.

[12] "Mobile Telemedicine", http://www.cs.jyu.fi/ai/mobile_telemedicine.html, 2002.

[13] Ministry of Health, "Kementerian Kesihatan Malaysia", http//www.moh.gov. my, 2004.

[14] Medijaring Sdn Bhd, "Netmyne e-health",www.health.com.my, 2004

[15] NEMA, "DICOM Strategy" http://medical.nema.org /dicom/ geninfo/ dicom_strategy / index.htm",2003.

[16] "Worldcare.com" www.worldcare.com.my/telehealth_index.html,2002.

[17] Robin Cover, "The cover pages", http://xml.coverpages.org/WAP20-Endorsements.html, 2001.

[18] 4i2i Communications Ltd, "H.263 Video coding tutorial", http://www.4i2i.com/h263_video_codec.htm, 2004.

[19] Eric Rescorla, "Claymore PureTLS", http://www.rtfm.com/puretls/,2003.

[20] Cryptix "Cryptix JCE" http://www.cryptix.org/products/jce/index.html,2002.

[21] Tony Johnson, Kevin Garwood, "Servlet How To", http://www.lip.pt/computing/docs/java/ jas/jas/documentation/howto/ servlet/ default.shtml, August 2000.

[22] Oracle, "JCB Advanced",http://otn.oracle.com/docs/tech/java/ servlets/htdocs/, 2004.

[23] Ray Patalano, "H.263 compresses video over IP", http:// www.nwfusion.com /news /tech/ 2002/ 0923 tech.html, September 2002.

[24] Kevin Hung and Yuan-Ting Zhang. 2003. "Implementation of a WAP-Based Telemedicine System for Patient Monitoring," IEEE Transactions on Information Technology in Biomedicine.Volume 7 Issue 2, 2003, pp.101-07.

[25] Kris Read and Frank Maurer. 2003. "Developing Mobile Wireless Application," IEEE Internet Computing. University of Calgary, Canada

[26] S. Petsas, D. Tzovaras, L. Makris, M.G.Strintzis. 2001. "WAP-Based Personalised Health Care Services," Proceedings of the 23rd Annual EMBS International Conference,Turkey. Aristotle University of Thessaloniki, Greece. pp 3536-3539.

[27] M Back, Th Norgall, M Rommel, Chr Zywietz. 1999. "Internet/Intranet Access to a Multimeida Cardiology Information System," IEEE Computers in Cardiology: 81-84

[28] Marcin Metterm, Dr. Robert Colomb. "WAP enabling existing HTML applications." University of Queensland, Australia.

[29] Pasi Leino. 2003. "Nokia XHTML for Cebit," Nokia Powerpoint Presentation Slides.

[30] Nicholas Foo. 2003. "Developing XHTML," Nokia Powerpoint Presentation Slides.

[31] Valerie Tai. 2003. "XHTML Content Creation Guidelines," Nokia Powerpoint Presentation Slides.

[32] Nokia. 2003. "Nokia WAP Phone Security,Version 3.3."

[33] Nokia. 2003. "Video and Streaming in Nokia Phones, Version 1.0"

[34] Nokia. 2003. "Video and Streaming Content Creation Guide, Version 1.0"

[35] Multimedia Malaysia. March2002. "Internet Case Study report," ITU

[36] H.M.Deitel,P.J.Deitel, T.R.Nieto, T.M. Lin,P. Sadhu. 2001. " XML How To Program", New Jersey,USA ; Prentice Hall.

[37] Laura Lemay.2003. "SAMs Teach Yourself Web Publishing with HTML and XHTML in 21 Days", Indiana,USA; SAMS.

[38] Jason Hunter and William Crawford. 2001. "Java Servlet Programming" USA; O'Reilly & Associates.

[39] H.M.Deitel,P.J.Deitel, S.E.Santry. 2002. "Advanced Java 2 Platform: How to Program", New Jersey,USA ; Prentice Hall.

[40] Andrea Steelman, Joel Murach.2003. "Murach's Java Servlets and JSP" USA; Mike Murach & Associates, Inc.

[41] Eric Rescorla.2001. "SSL and TLS –Designing and Building Secure Systems" USA, Addison Wesley.

[42] Elsen I, Hartung F,Horn U, Kampmann M, Peters L, "Streaming Technology in 3G Communication Systems", IEEE Computer, Volume 34 Issue 9, 2001, pp. 46-52.

APPENDIX A

Mobile Telemedicine Application files structure

```
Tomcat
    |------Webapps
            |-----login
                    |----- telemedicine.xhtml
                    |-----WEB-INF
                    |           |-----Web.xml
                    |           |-----Classes
                    |           |           |-----graph.java
                    |           |           |-----accesspg2.java
                    |           |           |-----access.java
                    |           |           |-----accesspg2.java
                    |           |           |-----access1.java
                    |           |           |-----access2.java
                    |           |           |-----ProtectedResource.java
                    |           |           |-----tracking.java
                    |           |-----lib
                    |                       |-----java jar files
                    |-----security
                            |-----errorpage.xhtml
                            |-----loginpage.xhtml
                            |-----streamoption.xhtml
                            |-----streaming.xhtml
                            |-----streaming1.xhtml
                            |-----hospitalmap.xhtml
                            |-----logout.jsp
                            |-----access
                            |           |-----access.xhtml
                            |           |-----form1.xhtml
                            |           |-----form2.xhtml
                            |           |-----graph.xhtml
                            |           |-----graphres.xhtml
                            |           |-----graphpulse.xhtml
                            |           |-----graphblood.xhtml
                            |
                            |-----cst
                            |           |-----text.css
                            |           |-----jpeg/gif files
                            |
                            |-----hospital
                                        |-----hos_fatimah.xhtml
                                        |-----hos_ipoh.xhtml
                                        |-----hppi.xhtrml
                                        |-----ipoh_special.xhtml
```

|-----kintamedic.xhtml
|-----pcmh.xhtml
|-----rambutan.xhtml

APPENDIX B

Apache Tomcat Web.xml coding

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE web-app
   PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.2//EN"
   "http://java.sun.com/j2ee/dtds/web-app_2_2.dtd">

<web-app>
            <display-name> A Secure Telemedicine Application </display-name>

     <servlet>
             <servlet-name> ProtectedResource </servlet-name>
             <servlet-class> ProtectedResource</servlet-class>
            </servlet>
            <servlet>
             <servlet-name> access </servlet-name>
             <servlet-class> access</servlet-class>
            </servlet>
            <servlet>
             <servlet-name> accesspg2 </servlet-name>
             <servlet-class> accesspg2</servlet-class>
            </servlet>
    <servlet>
             <servlet-name> access1 </servlet-name>
             <servlet-class> access1</servlet-class>
            </servlet>
   <servlet>
             <servlet-name> access2 </servlet-name>
             <servlet-class> access2</servlet-class>
            </servlet>
    <servlet>
             <servlet-name> graph </servlet-name>
             <servlet-class> graph</servlet-class>
            </servlet>
     <servlet>
             <servlet-name> tracking </servlet-name>
             <servlet-class> tracking</servlet-class>
            </servlet>


<servlet-mapping>
    <servlet-name> ProtectedResource  </servlet-name>
    <url-pattern>  /ProtectedResource </url-pattern>
 </servlet-mapping>
<servlet-mapping>
    <servlet-name> access  </servlet-name>
    <url-pattern> /access </url-pattern>
 </servlet-mapping>
<servlet-mapping>
    <servlet-name> accesspg2  </servlet-name>
    <url-pattern> /accesspg2 </url-pattern>
 </servlet-mapping>
<servlet-mapping>
    <servlet-name> access1  </servlet-name>
    <url-pattern>  /access1 </url-pattern>
 </servlet-mapping>
<servlet-mapping>
    <servlet-name> access2  </servlet-name>
    <url-pattern>  /access2 </url-pattern>
 </servlet-mapping>
<servlet-mapping>
    <servlet-name> tracking  </servlet-name>
    <url-pattern>  /tracking </url-pattern>
 </servlet-mapping>
<servlet-mapping>
```

```
      <servlet-name> graph  </servlet-name>
      <url-pattern>  /graph </url-pattern>
    </servlet-mapping>

 <!--
  <mime-mapping>
     <extension>xhtml</extension>
     <mime-type>text/html</mime-type>
  </mime-mapping>
 <mime-mapping>
    <extension>xhtml</extension>
    <mime-type>application/xhtml+xml</mime-type>
  </mime-mapping>
  <mime-mapping>
    <extension>xhtml</extension>
    <mime-type>application/vnd.wap.xhtml+xml</mime-type>
  </mime-mapping>
  <mime-mapping>
    <extension>wml</extension>
    <mime-type>text/vnd.wap.wml</mime-type>
  </mime-mapping>
  <mime-mapping>
    <extension>wmlc</extension>
    <mime-type>application/vnd.wap.wmlc</mime-type>
  </mime-mapping>
  <mime-mapping>
    <extension>wmls</extension>
    <mime-type>text/vnd.wap.wmls</mime-type>
  </mime-mapping>
  <mime-mapping>
    <extension>wmlscriptc</extension>
    <mime-type>application/vnd.wap.wmlscriptc</mime-type>
  </mime-mapping>
  <mime-mapping>
    <extension>wbmp</extension>
    <mime-type>image/vnd.wap.wbmp</mime-type>
  </mime-mapping>
 -->

  <security-constraint>
      <web-resource-collection>
         <web-resource-name> login</web-resource-name>
                        <url-pattern> /*</url-pattern>
                        <http-method>DELETE</http-method>
                        <http-method>PUT</http-method>
          <http-method>  GET  </http-method>
          <http-method>  POST  </http-method>
      </web-resource-collection>
     <user-data-constraint>
                        <transport-guarantee>CONFIDENTIAL</transport-guarantee>
            </user-data-constraint>
    <auth-constraint>
         <role-name> manager </role-name>
                        <role-name> user </role-name>
      </auth-constraint>

  </security-constraint>

   <login-config>
      <auth-method>
        FORM <!--BASIC, DIGEST, FORM, CLIENT-CERT -->
      </auth-method>
      <form-login-config>
         <form-login-page>
           /security/loginpage.xhtml
         </form-login-page>
         <form-error-page>
           /security/errorpage.xhtml
         </form-error-page>
      </form-login-config>
   </login-config>

 <security-role>
```

```
    <role-name> manager </role-name>
            <role-name>  user </role-name>
  </security-role>

  <session-config>
            <session-timeout>
                    10 <!--Minutes-->
            </session-timeout>
  </session-config>

  <welcome-file-list>
            <welcome-file>/telemedicine.xhtml</welcome-file>
  </welcome-file-list>

</web-app>
```

APPENDIX C

Java Servlets source code.

C.1 ProtectedResource.java

```
import java.io.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class ProtectedResource extends HttpServlet {

  public void doGet(HttpServletRequest req, HttpServletResponse res)
                    throws ServletException, IOException {
    res.setContentType("text/plain");
    PrintWriter out = res.getWriter();

    // Get the session
    HttpSession session = req.getSession();

    // Does the session indicate this user already logged in?
    Object done = session.getAttribute("logon.isDone");  // marker object

    if (done == null) {
      // No logon.isDone means she hasn't logged in.
      // Save the request URL as the true target and redirect to the login page.
      session.setAttribute("login.target",
                  HttpUtils.getRequestURL(req).toString());
      res.sendRedirect("/login/security/loginpage.xhtml");
      return;
    }

   res.sendRedirect("/telemedicine.xhtml");
  }
}
```

C.2 tracking.java

```
import java.io.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class tracking extends HttpServlet{

public void doGet(HttpServletRequest req, HttpServletResponse res)
                                    throws ServletException, IOException{

res.setContentType("text/html");
PrintWriter out=res.getWriter();
HttpSession session=req.getSession();


Integer count=(Integer)session.getAttribute("snoop.count");
if(count==null)
count=new Integer(1);
else
count=new Integer(count.intValue()+1);

session.setAttribute("snoop.count",count);
```

```
out.println("<HTML><HEAD><TITLE>Session info</TITLE></HEAD>");
out.println("<BODY><H3>Session </H3>");
out.println("You've visited this page" +count+((count.intValue()==1)? "time." :"times."));
out.println("<P>");
out.println("<H3> Here is your saved session data </H3>");

Enumeration enum=session.getAttributeNames();
while(enum.hasMoreElements()){
        String name=(String)enum.nextElement();
        out.println(name + ":" + session.getAttribute(name) + "<BR>");
        }

out.println("<H3>Here are some vital stats on your session: </H3>");
out.println("Session id: " +session.getId() +"<BR />");
out.println(" New session: " + session.isNew() +"<BR />");
out.println(" Timeout:" + session.getMaxInactiveInterval());
out.println("<I>(" + session.getMaxInactiveInterval()/60 +"minutes)</I><BR />");
out.println("Creation time:" + session.getCreationTime());
out.println("<I>("+new Date(session.getCreationTime()) +")</I><BR />");
out.println("Last access time: "+ session.getLastAccessedTime());
out.println("<I>("+new Date(session.getLastAccessedTime()) +")</I><BR />");
out.println("Requested session ID from cookie:"+ req.isRequestedSessionIdFromCookie()+"<BR />");
out.println("Requested session ID from URL:"+ req.isRequestedSessionIdFromURL()+"<BR />");
out.println("Requested session ID valid"+ req.isRequestedSessionIdValid()+"<BR />");

out.println("</BODY></HTML>");
}
}
```

C.3 graph.java

```
import java.awt.*;
import java.awt.image.*;
import java.io.*;
import javax.imageio.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
import java.net.*;
import java.util.Enumeration;
import javachart.chart.*;
import javachart.utility.*;
import java.lang.*;

public class graph extends HttpServlet{
   Connection theConnection;
    private ServletConfig config;

public void init()throws ServletException{}
public void doGet(HttpServletRequest incoming, HttpServletResponse outgoing)
 {
HttpSession session = incoming.getSession(true);
ServletContext context=getServletContext();
String username =(String)context.getAttribute("user.com");

String WantedTable= incoming.getParameter("table");
String WantedColumn= incoming.getParameter("column");
String WantedTime= incoming.getParameter("time");

try{

outgoing.setContentType("image/jpeg");
ServletOutputStream out = outgoing.getOutputStream();
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
theConnection = DriverManager.getConnection("jdbc:odbc:db1","","");

Statement theStatement=theConnection.createStatement();
```

```
ResultSet theResult=theStatement.executeQuery("select * from UserDBG WHERE UserDB.LoginName = UserDBG.Client");

ArrayList ValuesY = new ArrayList();
ArrayList ValuesX = new ArrayList();
double temp;

while(theResult.next()){
        String current= theResult.getString(WantedColumn);
    ValuesY.add(current);
    String x=theResult.getString(WantedTime);
    ValuesX.add(x);
}

RenderedImage myimg = new BufferedImage(170,160,BufferedImage.TYPE_INT_RGB);
Graphics g = ((BufferedImage)myimg).createGraphics();

//create the chart
LineChart chart = new LineChart("Temperature");
double[] data = new double[ValuesY.size()];
double[] labels= new double[ValuesX.size()];

for (int k=0; k<ValuesY.size();k++)
{data[k]= Double.parseDouble((String)ValuesY.get(k));
 labels[k]=Double.parseDouble((String)ValuesX.get(k));
}
chart.addDataset("CST",labels,data);
//chart.getBackground().setTitleString("Temperature");
chart.getBackground().getGc().setFillColor(Color.lightGray);
chart.resize(170,160);
chart.getXAxis().setGridVis(true);
chart.getYAxis().setGridVis(true);
chart.getXAxis().setTitleString("Time");
chart.getYAxis().setTitleString("Celcious");
chart.drawGraph(g);
//encode the image and send it
ImageIO.write(myimg,"jpeg",out);
out.close();
 theResult.close();//Close the result set
 theStatement.close();//Close statement
 theConnection.close(); //Close database Connection
}
catch (Exception e) {}
}
}
```

C.4 access.java

```
import java.io.*;
import java.util.Enumeration;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
import java.net.*;

public class access extends HttpServlet{
   Connection theConnection;
   private ServletConfig config;

public void init()throws ServletException{}
public void doPost(HttpServletRequest req, HttpServletResponse res)
throws ServletException, IOException {

  HttpSession session = req.getSession(true);
  String username=req.getParameter("username");
  String type=req.getParameter("access_type");
  boolean userrole=req.isUserInRole("manager");
  String login=req.getRemoteUser();
  res.setContentType("text/html");
  PrintWriter out = res.getWriter();
```

```
  ServletContext context=getServletContext();
 context.setAttribute("user.com", username);

if(type.equals("Update")){
                        if(userrole==true){
                        res.sendRedirect("/login/security/access/form1.xhtml");
                        }else{
                        out.println("Sorry,you are not authorize to proceed");
                        }
}

if(type.equals("View")){
  out.println("<html><head><title>Database</title>");
  out.println("</head>");
  out.println("<body bgColor=blanchedalmond text=#008000 topMargin=0>");
  out.println("<p align=center><font face=Helvetica color=fuchsia><BIG>Database listing</BIG></FONT></P>");
  out.println("<p align=center>");
  out.println("<table align=center border=1 cellPadding=1 cellSpacing=1 width=\"75%\">");

 try{

    //Loading Sun's JDBC ODBC Driver
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
   theConnection = DriverManager.getConnection("jdbc:odbc:db1","","");
  Statement theStatement=theConnection.createStatement();
  // Statement theStatement1=theConnection.createStatement();

  ResultSet theResult=theStatement.executeQuery("select * from UserDB where LoginName='"+username+"'");

 //Fetch all the records and print in table
   while(theResult.next()){

   out.println("<TR><TD>Account ID</TD>");
   out.println("<TD>" + theResult.getString(1) + "</TD></TR>");

   out.println("<TR><TD>Name</TD>");
   out.println("<TD>" + theResult.getString(3) + "</TD></TR>");

   out.println("<TR><TD>Attending Doctor</TD>");
   out.println("<TD>" + theResult.getString(2) + "</TD></TR>");

   out.println("<TR><TD>E-mail</TD>");
   out.println("<TD>" + theResult.getString(6) + "</TD></TR>");

    out.println("<TR><TD>Registration Number</TD>");
   out.println("<TD>" + theResult.getString(7) + "</TD></TR>");

   out.println("<TR><TD>Patient Address</TD>");
   out.println("<TD>" + theResult.getString(8) + "</TD></TR>");

   out.println("<TR><TD>Telephone number</TD>");
   out.println("<TD>" + theResult.getString(9) + "</TD></TR>");

   out.println("<TR><TD>Postcode</TD>");
   out.println("<TD>" + theResult.getString(10) + "</TD></TR>");

   out.println("<TR><TD>NRIC number</TD>");
   out.println("<TD>" + theResult.getString(11) + "</TD></TR>");

   out.println("<TR><TD>Date of Birth</TD>");
   out.println("<TD>" + theResult.getString(12) + "</TD></TR>");

   out.println("<TR><TD>Age</TD>");
   out.println("<TD>" + theResult.getString(13) + "</TD></TR>");

   out.println("<TR><TD>Gender</TD>");
   out.println("<TD>" + theResult.getString(14) + "</TD></TR>");

   out.println("<TR><TD>Marital Status</TD>");
   out.println("<TD>" + theResult.getString(15) + "</TD></TR>");

   out.println("<TR><TD>Race</TD>");
   out.println("<TD>" + theResult.getString(16) + "</TD></TR>");
```

```
   out.println("<TR><TD>Nationality</TD>");
   out.println("<TD>" + theResult.getString(17) + "</TD></TR>");

   out.println("<TR><TD>Occupation</TD>");
   out.println("<TD>" + theResult.getString(18) + "</TD></TR>");
 }
   theResult.close();//Close the result set
  theStatement.close();//Close statement
  theConnection.close(); //Close database Connection

  }catch(Exception e){
   out.println(e.getMessage());//Print trapped error.
  }
out.println("</table></p></br>");
out.println("<ol><li><a href=/login/accesspg2>Continue page 2</a></li>");
out.println("<li><a href=/login/security/access/graph.xhtml"+">Temperature readings</a></li>");
out.println("<li><a href="+"/login/security/access/graphpulse.xhtml"+">Pulse readings</a></li>");
out.println("<li><a href="+"/login/security/access/graphblood.xhtml"+">Blood Presure readings</a></li>");
out.println("<li><a href="+"/login/security/access/graphres.xhtml"+">Respiration readings</a></li></ol>");
out.println("<P> </P></font></body></html>");
 }}

public void destroy(){
}
}
```

C.5 accesspg2.java

```
import java.io.*;
import java.util.Enumeration;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
import java.net.*;

 public class accesspg2 extends HttpServlet{
    Connection theConnection;
    private ServletConfig config;

public void init()throws ServletException{}

public void doGet(HttpServletRequest req, HttpServletResponse res)
throws ServletException, IOException {

HttpSession session = req.getSession(true);

ServletContext context=getServletContext();
String username =(String)context.getAttribute("user.com");

res.setContentType("text/html");
PrintWriter out = res.getWriter();

   out.println("<html><head><title>Database Page 2</title>");
   out.println("</head>");
   out.println("<body bgColor=blanchedalmond text=#008000 topMargin=0>");
   out.println("<p align=center><font face=Helvetica color=fuchsia><BIG>Database listing</BIG></FONT></P>");

   out.println("<p align=center>");
   out.println("<table align=center border=1 cellPadding=1 cellSpacing=1 width=\"75%\">");

    try{

    //Loading Sun's JDBC ODBC Driver
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    //Connect to Microsoft Access Database
    theConnection = DriverManager.getConnection("jdbc:odbc:db1","","");
     Statement theStatement=theConnection.createStatement();
    ResultSet theResult=theStatement.executeQuery("select * from  UserDB where LoginName='"+username+"'");
```

```
while(theResult.next()){
out.println("<TR><TD>Admission_date</TD>");
out.println("<TD>" + theResult.getString(20) + "</TD></TR>");

out.println("<TR><TD>Financial_class</TD>");
out.println("<TD>" + theResult.getString(21) + "</TD></TR>");

out.println("<TR><TD>Time</TD>");
out.println("<TD>" + theResult.getString(22) + "</TD></TR>");

out.println("<TR><TD>Ward</TD>");
out.println("<TD>" + theResult.getString(23) + "</TD></TR>");

out.println("<TR><TD>Room</TD>");
out.println("<TD>" + theResult.getString(24) + "</TD></TR>");

out.println("<TR><TD>Bed</TD>");
out.println("<TD>" + theResult.getString(25) + "</TD></TR>");

out.println("<TR><TD>Next of kin</TD>");
out.println("<TD>" + theResult.getString(26) + "</TD></TR>");

out.println("<TR><TD>Next of kin relationship</TD>");
out.println("<TD>" + theResult.getString(27) + "</TD></TR>");

out.println("<TR><TD>Next of kin address</TD>");
out.println("<TD>" + theResult.getString(28) + "</TD></TR>");

out.println("<TR><TD>Next of kin postcode</TD>");
out.println("<TD>" + theResult.getString(29) + "</TD></TR>");

out.println("<TR><TD>Next of kin tel</TD>");
out.println("<TD>" + theResult.getString(30) + "</TD></TR>");

out.println("<TR><TD>Guarantor</TD>");
out.println("<TD>" + theResult.getString(31) + "</TD></TR>");

out.println("<TR><TD>Guarantor relationship</TD>");
out.println("<TD>" + theResult.getString(32) + "</TD></TR>");

out.println("<TR><TD>Guarantor NRIC</TD>");
out.println("<TD>" + theResult.getString(33) + "</TD></TR>");

out.println("<TR><TD>Guarantor address</TD>");
out.println("<TD>" + theResult.getString(34) + "</TD></TR>");

out.println("<TR><TD>Guarantor postcode</TD>");
out.println("<TD>" + theResult.getString(35) + "</TD></TR>");

out.println("<TR><TD>Employer</TD>");
out.println("<TD>" + theResult.getString(36) + "</TD></TR>");

out.println("<TR><TD>Employer add</TD>");
out.println("<TD>" + theResult.getString(37) + "</TD></TR>");

out.println("<TR><TD>Employer postcode</TD>");
out.println("<TD>" + theResult.getString(38) + "</TD></TR>");

out.println("<TR><TD>Guarantor tel</TD>");
out.println("<TD>" + theResult.getString(39) + "</TD></TR>");

out.println("<TR><TD>Remarks</TD>");
out.println("<TD>" + theResult.getString(40) + "</TD></TR>");

out.println("<TR><TD>Ambulence</TD>");
out.println("<TD>" + theResult.getString(41) + "</TD></TR>");

out.println("<TR><TD>Discharge</TD>");
out.println("<TD>" + theResult.getString(42) + "</TD></TR>");

out.println("<TR><TD>Deposit Paid</TD>");
out.println("<TD>" + theResult.getString(43) + "</TD></TR>");
```

```
 }
  theResult.close();//Close the result set
  theStatement.close();//Close statement
  theConnection.close(); //Close database Connection

  }catch(Exception e){
   out.println(e.getMessage());//Print trapped error.
  }

out.println("</table></p></br>");
out.println("<ol><li><a href=/login/security/access/graph.xhtml"+">Temperature readings</a></li>");
out.println("<li><a href="+"/login/security/access/graphpulse.xhtml"+">Pulse readings</a></li>");
out.println("<li><a href="+"/login/security/access/graphblood.xhtml"+">Blood Presure readings</a></li>");
out.println("<li><a href="+"/login/security/access/graphres.xhtml"+">Respiration readings</a></li>");
out.println("<li><a href=/login/telemedicine.xhtml>Back to Main Page</a></li></ol>");
out.println("<P> </P></font></body></html>");
 }
public void destroy(){
}
}
```

C.6 access1.java

```
import java.io.*;
import java.util.Enumeration;
import java.text.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
import java.net.*;

public class access1 extends HttpServlet{
     Connection connection;
      private ServletConfig config;

public void init()throws ServletException{}
  protected void doPost(HttpServletRequest req, HttpServletResponse res)
throws ServletException, IOException {
 Statement stmt=null;

HttpSession session = req.getSession(true);
 res.setContentType("text/html");
 PrintWriter out = res.getWriter();
ServletContext context=getServletContext();
String username =(String)context.getAttribute("user.com");
String accno=req.getParameter("accno");
String regno=req.getParameter("regno");
String patname=req.getParameter("patname");
String pw=req.getParameter("pw");
String attdr=req.getParameter("attdr");
String add=req.getParameter("add");
String pstcode=req.getParameter("pstcode");
String telno=req.getParameter("telno");
String nricno=req.getParameter("nricno");
String birthd=req.getParameter("birthd");
String age=req.getParameter("age");
String gender=req.getParameter("gender");
String marsta=req.getParameter("marsta");
String race=req.getParameter("race");
String national=req.getParameter("national");
String occ=req.getParameter("occ");

try{
  //Loading Sun's JDBC ODBC Driver
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    connection = DriverManager.getConnection("jdbc:odbc:db1","","");

stmt=connection.createStatement();
int count= stmt.executeUpdate("UPDATE UserDB SET Account ID ="
```

```
                                +accno+"',Registration No ='"
                                +regno+"',Name ='"
                                +patname+"',Password = '"
                                +pw+"',Atten-Dr = '"
                                +attdr+"',Patient Address ='"
                                +add+"',Postcode = '"
                                +pstcode+"',Tel No = '"
                                +telno+"',NRIC = '"
                                +nricno+"',              dbDOB = '"
                                +birthd+"',Age ='"
                                +age+"',dbGender ='"
                                +gender+"',dbMaritalStatus = '"
                                +marsta+"',Race = '"
                                +race+"',dbNationality = '"
                                +national+"',Occupation = '"
                                +occ+"' WHERE LoginName='"+username+"'");

  out.println("<head><title>Update Completed</title>");
  out.println("</head><body>");
  out.println(" <p>Update completed </p><br />");
  out.println(" <a href=access2>Click here to proceed </a>");
  out.println(" <a href=telemedicine.xhtml>Back to Main Page</a>");
   out.println( "</body></html>");
  out.close();
  stmt.close();

  }catch(Exception e){
   out.println(e.getMessage());//Print trapped error.
  out.println("<br/>");
  e.printStackTrace();
  out.println("<title>Error</title>");
  out.println("</head>");
  out.println("<body><p>Database error occurred.");
  out.println("Try again later </p></body></html>");
  out.close();
  }
}
public void destroy(){
try{
   connection.close(); //Close database Connection
  }catch(SQLException sqlException){
            sqlException.printStackTrace();}
}}
```

C.7 access2.java

```
import java.io.*;
import java.util.Enumeration;
import java.text.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
import java.net.*;

public class access2 extends HttpServlet{
     Connection connection;
      private ServletConfig config;

public void init()throws ServletException{}

protected void doPost(HttpServletRequest req, HttpServletResponse res)
throws ServletException, IOException {

 Statement stmt=null;

HttpSession session = req.getSession(true);
 res.setContentType("text/html");
 PrintWriter out = res.getWriter();
ServletContext context=getServletContext();
```

```
String username =(String)context.getAttribute("user.com");
String admindate1=req.getParameter("admindate");
String finclass1=req.getParameter("finclass");
String time=req.getParameter("time");
String ward=req.getParameter("ward");
String room=req.getParameter("room");
String bed=req.getParameter("bed");
String kin=req.getParameter("kin");
String kinrel=req.getParameter("kinrel");
String kin_add=req.getParameter("kin_add");
String kinpostc=req.getParameter("kinpostc");
String kintel=req.getParameter("kintel");
String guaname=req.getParameter("guaname");
String guarel=req.getParameter("guarel");
String guanric=req.getParameter("guanric");
String guaadd=req.getParameter("guanadd");
String guapost=req.getParameter("guapost");
String guatel=req.getParameter("guatel");
String empname=req.getParameter("empname");
String emptel=req.getParameter("emptel");
String empadd=req.getParameter("empadd");
String emppost=req.getParameter("emppost");
String remark=req.getParameter("remark");
String amb=req.getParameter("amb");
String discharge=req.getParameter("discharge");
String deppaid=req.getParameter("deppaid");


try{

 //Loading Sun's JDBC ODBC Driver
   Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
   connection = DriverManager.getConnection("jdbc:odbc:db1","","");

 stmt=connection.createStatement();
 int count= stmt.executeUpdate("UPDATE UserDB SET Admission_date='"
                                                  +admindate1+"', Financial_class='"
                                                  +finclass1+"',Time='"
                                                  +time+"',Ward='"
                                                  +ward+"', Room='"
                                                  +room+"',Bed='"
                                                  +bed+"', Next of kin='"
                                                  +kin+"',Kin_relationship='"
                                                  +kinrel+"',Kin_Address='"
                                                  +kin_add+"', Kin_postcode='"
                                                  +kinpostc+"', Kin_tel_no='"
                                                  +kintel+"', Guarantor='"
                                                  +guaname+"',Gua_relationship='"
                                                  +guarel+"',Gua_nric='"
                                                  +guanric+"',Gua_Address ='"
                                                  +guaadd+"',Gua_Postcode='"
                                                  +guapost+"',Gua_tel='"
                                                  +guatel+"',Employer='"
                                                  +empname+"',Emp_tel ='"
                                                  +emptel+"',Emp_address ='"
                                                  +empadd+"',Emp_postcode ='"
                                                  +emppost+"',Remarks ='"
                                                  +remark+"',Ambulence='"
                                                  +amb+"',Discharge ='"
                                                  +discharge+"',Deposit paid ='"
                                                  +deppaid+"' WHERE LoginName='"+username+"'");

out.println("<head><title>Update Completed</title>");
 out.println("</head><body>");
 out.println(" <p>Update completed </p>");
 out.println(" <a href=telemedicine.xhtml>Click here to proceed </a>");
  out.println( "</body></html>");
 out.close();
 stmt.close();

 }catch(Exception e){
  out.println(e.getMessage());//Print trapped error.
 out.println("<br/>");
```

```
  e.printStackTrace();
  out.println("<title>Error</title>");
  out.println("</head>");
  out.println("<body><p>Database error occurred.");
  out.println("Try again later </p></body></html>");
  out.close();
  }
}

public void destroy(){
try{
   connection.close(); //Close database Connection
  }catch(SQLException sqlException){
            sqlException.printStackTrace();}
}
}
```

APPENDIX D

XHTML source codes

D.1 telemedicine.xhtml

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<?xml version="1.0"?><HTML xmlns="http://www.w3.org/1999/xhtml">
<HEAD><TITLE>Mobile Telemedicine</TITLE>
<META http-equiv=Content-Type content="text/html; charset=windows-1252"><LINK
href="security/cst/text.css" type=text/css rel=stylesheet>
<STYLE type=text/css></STYLE>
</HEAD>
<BODY><IMG alt="toplogo" src="security/cst/toplogo.jpg">
<TABLE>
 <TBODY>
 <TR>
  <TD width=55><IMG alt=Icon src="security/cst/healthrecord.jpg"></TD>
  <TD width=115><A class=head style="COLOR: #702e2e"
   href="security/access/access.xhtml">Health Records</A>
   <DIV class=sub style="LINE-HEIGHT: 8px">View & update </DIV></TD></TR>
 <TR>
  <TD width=55><IMG alt=Icon src="security/cst/streaming.jpg"></TD>
  <TD width=115><A class=head style="COLOR: #702e2e"
   href="security/streamoption.xhtml">Streaming Video</A>
   <DIV class=sub style="LINE-HEIGHT: 8px">Video & Audio Feed</DIV></TD></TR>
 <TR>
  <TD width=55><IMG alt=Icon src="security/cst/Hos_info.jpg"></TD>
  <TD width=115><A class=head style="COLOR: #702e2e"
   href="security/hospitalmap.xhtml">Hospital Info</A>
   <DIV class=sub style="LINE-HEIGHT: 8px">Location and info of hospitals</DIV></TD></TR>
<TR>
  <TD width=55><IMG alt=Icon src="security/cst/Conn_info.jpg"></TD>
  <TD width=115><A class=head style="COLOR: #702e2e"
   href="/login/tracking">Session info</A>
   <DIV class=sub style="LINE-HEIGHT: 8px">Connection information</DIV></TD></TR>

</TBODY>
</TABLE>
<IMG src="security/cst/footer.jpg"><br />
<a href="security/logout.jsp?logoff=true">Log out</a>
</BODY>
</HTML>
```

D.2 loginpage.xhtml

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<?xml version="1.0"?><HTML xmlns="http://www.w3.org/1999/xhtml">
<HTML>
<TITLE>Login</TITLE>
<BODY>
<FORM METHOD=POST ACTION=j_security_check>
<CENTER>
<TABLE BORDER=0>
<TR><TD COLSPAN=2>
<P ALIGN=center>
Welcome!  Please enter your Name<br>
 and Password to log in.
</TD></TR>
<TR><TD>
```

```
<P ALIGN=right><B>Name:</B>
</TD>
<TD>
<P><INPUT TYPE=TEXT NAME="j_username" VALUE="" SIZE=12>
</TD></TR>
<TR><TD>
<P ALIGN=RIGHT><B>Password:</B>
</TD>
<TD>
<P><INPUT TYPE=PASSWORD NAME="j_password" VALUE="" SIZE=12>
</TD></TR>
<TR><TD COLSPAN=2>
<CENTER>
<INPUT TYPE=submit VALUE="  OK  ">
</CENTER>
</TD></TR>
</TABLE>
</FORM>
</BODY></HTML>
```

D.3 errorpage.xhtml

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<?xml version="1.0"?>
<HTML xmlns="http://www.w3.org/1999/xhtml">
<head>
<TITLE>Login Denied</TITLE>
</head>
<BODY>
Sorry, your login was denied.
Please hit the Back button to try again.
</BODY>
</HTML>
```

D.4 hospitalmap.xhtml

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<?xml version="1.0"?><HTML xmlns="http://www.w3.org/1999/xhtml">
<head><title>Hospital Information</title>
<link href="cst/text.css" type=text/css rel=stylesheet>
<style type=text/css></style>
</head>
<body bgcolor="#f5deb3">
<table>
 <tbody>
 <tr>
   <td width=170><a3><b><center>Hospitals in Ipoh </center> </b></a3> </td>
 </tr>
<tr>
   <td width=170><a class=head style="COLOR: #702e2e"
    href="hospital/hos_ipoh.xhtml"><font size="2">1. Hospital Besar Ipoh</font></a> </td>
 </tr>
<tr>
   <td width=170><a class=head style="COLOR: #702e2e"
    href="hospital/hos_fatimah.xhtml"><font size="2">2. Hospital Fatimah</font></a> </td>
 </tr>
<tr>
   <td width=170><A class=head style="COLOR: #702e2e"
    href="hospital/kintamedic.xhtml"><font size="2">3. Kinta Medical Center</font></a> </td>
 </tr>
<tr>
   <td width=170><A class=head style="COLOR: #702e2e"
    href="hospital/ipoh_special.xhtml"><font size="2">4. Ipoh Specialist Centre</font></a> </td>
 </tr>
<tr>
```

```
    <td width=170><A class=head style="COLOR: #702e2e"
    href="hospital/hppi.xhtml"><font size="2">5. Hospital Pantai Putri Ipoh</font></a> </td>
  </tr>
 <tr>
    <td width=170><A class=head style="COLOR: #702e2e"
    href="hospital/pcmh.xhtml"><font size="2">6. Perak Chinese Maternity Hospital</font></a> </td>
  </tr>
 <tr>
    <td width=170><A class=head style="COLOR: #702e2e"
    href="hospital/rambutan.xhtml"><font size="2">7. Hospital Bahagia Tanjung Rambutan </font></a> </td>
  </tr>

 </tbody>
</table>
<img src="cst/footer.jpg">
</body></html>
```

D.5 streamoption.xhtml

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<?xml version="1.0"?><HTML xmlns="http://www.w3.org/1999/xhtml">
<head><title>Streaming Media</title>
</head>
<html>
<body>
<p> Please select if you want to stream using mobile or computer
<a href="streaming.xhtml">Computer<a><br />
<a href="streaming1.xthml">Mobile phone<a><br /><p>
</body>
</html>
```

D.6 streaming.xhtml

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<?xml version="1.0"?><HTML xmlns="http://www.w3.org/1999/xhtml">
<head><title>Streaming Media</title>
</head>
<html>
<body>
<body bgcolor="#b0e0e6">
<object CLASSID="clsid:02BF25D5-8C17-4B23-BC80-D3488ABDDC6B" WIDTH="200"HEIGHT="150"
CODEBASE="http://www.apple.com/qtactivex/qtplugin.cab">
<param name="SRC" value="aa002-3G.3gp">
<param name="AUTOPLAY" value="false">
<param name="CONTROLLER" value="true">
<param name="scale" value="tofit">
<embed SRC="aa002-3G.3gp" WIDTH="200" HEIGHT="150" AUTOPLAY="false" CONTROLLER="true"
type="video/quicktime" scale="tofit" PLUGINSPAGE="http://www.apple.com/quicktime/download/">
</embed>
</object>
</body>
</html>
```

D.7 streaming1.xhtml

```
<?xml version="1.0"?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD XHTML Mobile 1.0//EN"
  "http://www.wapforum.org/DTD/xhtml-mobile10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
<head><title>Streaming Media</title>
</head>
<html>
<body>
<body bgcolor="#b0e0e6">
<p>
<a href="rtsp://localhost/aa002-3G.3gp">Start video</a>
</p>
</body>
</html>
```

D.8 access.xhtml

```
<?xml version="1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Accessing database</title>
</head>

<body>
<form action="/login/access" method ="post">
<h3> Entering Health Records </h3>
<p> Please type your username:<br />
<input type="text" name="username" /> <br />
</p>
<p>Please select one: <br />
<input type="radio" name="access_type" value="View" /> View<br/>
<input type="radio" name="access_type" value="Update" /> Update<br/>
Note: Update option is only available for administrator <br/>
</p>
<input type="submit" value="Proceed" />

</form>
</body>
</html>
```

D.9 form1.xhtml

```
<?xml version="1.0" ?>
<!DOCTYPE html PUBLIC "-//WAPFORUM//DTD XHTML Mobile
1.0//EN" "http://www.wapforum.org/DTD/xhtml-mobile10.dtd">
<html>
<head>
<title>Form Part 1</title>
</head>

<body>

</p>
<h2>Form 1</h2>
<form action="/login/access1" method ="post">
<p>

<label>Account No.:</label>
<input type="text" name="accno" size="10" maxlength="20" /><br />
<label>Registration No. :</label>
<input type="text" name="regno" size="10" maxlength="20" /><br />
<label>Patient Name:</label>
<input type="text" name="patname" size="10" maxlength="20"  value="username"/><br />
<label>Password:</label>
<input type="password" name="pw"  size="10" maxlength="20" /><br />
<label>Attending Doctor:</label>
```

```
<input type="text" name="attdr" size="10" maxlength="20" value="Dr."/><br />
<label>Address:</label>
<textarea name="add" rows="3" cols="10">Address</textarea><br />
<label>Postcode:</label>
<input type="text" name="pstcode" size="8" maxlength="10" /><br />
<label>Tel no.: </label>
<input type="text" name="telno" size="8" maxlength="10" /><br />
<label>NRIC No.:</label>
<input type="text" name="nricno" size="10" maxlength="20" value="800000145000"/><br />
<label>Birthdate:</label>
<input type="text" name="birthd" size="8" maxlength="10" value="23/05/81"/>
<label>Age:</label>
<input type="text" name="age" size="3" maxlength="5" /><br />
<label>Sex: </label>
<input type="radio" name="gender" value="F" checked="checked" />Female
<input type="radio" name="gender" value="M" /> Male <br />
<label>Marital Status:</label><br />
<input type="checkbox" name="marsta" value="single" checked="checked" />Single<br />
<input type="checkbox" name="marsta" value="married" />Married<br />
<input type="checkbox" name="marsta" value="divorce" />Divorce<br />
<label>Race:</label>
<input type="text" name="race" size="10" maxlength="20" value="Malay"/><br />
<label>Nationality:</label>
<input type="text" name="national" size="10" maxlength="20" value="Malaysian"/><br />
<label>Occupation:</label>
<input type="text" name="occ" size="10" maxlength="20" /><br />
<input type="submit" name="Submit" value="Submit" />
<input type="reset" name="Reset" value="Clear" />
</p>
</form>

<p><a href="form2.xhtml">Next</a></p>
<p><a href="../../telemedicine.xhtml">Home</a></p>
</body>
</html>
```

D.10 form2.xhtml

```
<?xml version="1.0" ?>
<!DOCTYPE html PUBLIC "-//WAPFORUM//DTD XHTML Mobile
1.0//EN" "http://www.wapforum.org/DTD/xhtml-mobile10.dtd">

<html>
<head>
<title>Form Part 2</title>
</head>

<body>

</p>
<h3>Status of admission:</h3>
<form action="/login/access2" method ="post">

<p>
<label>Admission date</label>
<input type="text" name="admindate" size="10" maxlength="20" /><br />
<label>Financial Class </label>
<input type="text" name="finclass" size="10" maxlength="20" value="Selfpay"/><br />
<label>Time </label>
<input type="text" name="time" size="10" maxlength="20" />
<label>Ward</label>
<input type="text" name="ward" size="3" maxlength="20" /><br />
<label>Room</label>
<input type="text" name="room" size="3" maxlength="20" />
<label>Bed</label>
<input type="text" name="bed" size="3" maxlength="20" /><br />
<h3>Incase of emergency contact:</h3>
<label>Next of kin:</label>
<input type="text" name="kin" size="20" maxlength="20" /><br />
```

```
<label>Relationship: </label>
<input type="text" name="kinrel" size="20" maxlength="20" value="Father"/><br />
<label> Address:</label>
<textarea name="kin_add" rows="3"cols="10">Address</textarea><br />
<label>Postcode:</label>
<input type="text" name="kinpostc" size="8" maxlength="10" /><br />
<label>Tel no. :</label>
<input type="text" name="kintel" size="8" maxlength="10" /><br />
<h3>Guarantor:</h3>
<label>Guarantor Name:</label>
<input type="text" name="guaname" size="10" maxlength="20" /><br />
<label>Relationship: </label>
<input type="text" name="guarel" size="20" maxlength="20" /><br />
<label>NRIC No:</label>
<input type="text" name="guanric"  size="10" maxlength="5" /><br />
<label> Address: </label>
<textarea name="guaadd" rows="3"cols="10">Address</textarea><br />
<label>Postcode:</label>
<input type="text" name="guaadd" size="8" maxlength="10" /><br />
<label>Tel no.: </label>
<input type="text" name="guatel" size="8" maxlength="10" /><br />
<h3>Employers particulars:</h3>
<label>Name:</label>
<input type="text" name="empname" size="10" maxlength="20" /><br />
<label>Tel no.: </label>
<input type="text" name="emptel" size="8" maxlength="10" /><br />
<label> Address: </label>
<textarea name="empadd" rows="3"cols="10">Address</textarea><br />
<label>Postcode:</label>
<input type="text" name="emppost" id="feld1" size="8" maxlength="10" /><br />
<label> Remarks: </label>
<textarea name="remark" rows="3"cols="10">Remarks</textarea><br />
<h3>Others:</h3>
<label>Ambulence:</label>
<input type="checkbox" name="amb" value="yes" checked="checked" />Yes
<input type="checkbox" name="amb" value="no" />No<br />

<label>Discharge:</label>
<input type="checkbox" name="discharge" value="yes" checked="checked" />Yes
<input type="checkbox" name="discharge" value="no" />No<br />

<label>Deposit paid:</label>
<input type="text" name="deppaid" size="8" maxlength="10" value="RM"/><br />
<input type="submit" name="Submit" value="Submit" />
<input type="reset" name="Reset" value="Clear" />
</p>
</form>
<p><a href="../../telemedicine.xhtml">Home</a></p>
</body>
</html>
```

D.11 graph.xhtml

```
<?xml version="1.0"?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD XHTML Mobile 1.0//EN"
  "http://www.wapforum.org/DTD/xhtml-mobile10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Accessing database-GraphData</title>
</head>
<body>
<p>
<center><h3> Health Records Data</h3>
Temperature graph<br />
<img src="http://localhost:8080/login/graph?table=UserDBG&column=Temperature&time=Time"><br />
</center></p>
<a href="../telemedicine.xhtml">Back to Mainpage</a>
</body>
</html>
```

D.12 graphres.xhtml

```
<?xml version="1.0"?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD XHTML Mobile 1.0//EN"
  "http://www.wapforum.org/DTD/xhtml-mobile10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Accessing database-GraphData</title>
</head>
<body>
<p>
<center><h3> Health Records Data</h3>
Temperature graph<br />
<img src="http://localhost:8080/login/graph?table=UserDBG&column=Respiration&time=Time"><br />
</center></p>
<a href="../telemedicine.xhtml">Back to Mainpage</a>
</body>
</html>
```

D.13 graphpulse.xhtml

```
<?xml version="1.0"?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD XHTML Mobile 1.0//EN"
  "http://www.wapforum.org/DTD/xhtml-mobile10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Accessing database-GraphData</title>
</head>
<body>
<p>
<center><h3> Health Records Data</h3>
Temperature graph<br />
<img src="http://localhost:8080/login/graph?table=UserDBG&column=Pulse&time=Time"><br />
</center></p>
<a href="../telemedicine.xhtml">Back to Mainpage</a>
</body>
</html>
```

D.14 graphblood.xhtml

```
<?xml version="1.0"?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD XHTML Mobile 1.0//EN"
  "http://www.wapforum.org/DTD/xhtml-mobile10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Accessing database-GraphData</title>
</head>
<body>
<p>
<center><h3> Health Records Data</h3>
Temperature graph<br />
<img src="http://localhost:8080/login/graph?table=UserDBG&column=Blood Pressure&time=Time"><br />
</center></p>
<a href="../telemedicine.xhtml">Back to Mainpage</a>
</body>
</html>
```

D.15 hos_ipoh.xhtml

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<?xml version="1.0"?><HTML xmlns="http://www.w3.org/1999/xhtml">
<head><title>Hospital Information</title>
<link href="cst/funky.css" type=text/css rel=stylesheet>
<style type=text/css></style>
</head>

<body bgcolor="#f5deb3">
<font size="3"><b>1. Hospital Besar Ipoh</b></font>
<br /> <br />
<table>
 <tbody>
<tr>
   <td width="35"><p>Direction</p></td>
  <td width="125"><a href="pic1.xhtml" > Click here for map</a></td>
</tr>
<tr>
   <td width="35"><p>Address</p></td>
  <td width="125"><p>Jalan Hospital, 31400 Ipoh.</p></td>
</tr>
<tr>
   <td width="35"><p>Telephone</p></td>
  <td width="125"><p><a href="wtai://wp/mc;055455777">05-253 3333 </a>;05-253 6040</p></td>
</tr>
<tr>
   <td width="35"><p>Fax</p></td>
  <td width="125"><p>05-253 1541</p></td>
</tr>
</tbody>
</table>
<p><a href="../hospitalmap.xhtml">Back</a> |<a href="../../telemedicine.xhtml">Back to Main Page</a></p>
</body>
</html>
```

APPENDIX E

CSS source code

```
BODY {
	PADDING-RIGHT: 0px; PADDING-LEFT: 0px; PADDING-BOTTOM: 0px; MARGIN: 0px; BORDER-TOP-
STYLE: none; PADDING-TOP: 0px; BORDER-RIGHT-STYLE: none; BORDER-LEFT-STYLE: none; BACKGROUND-
COLOR: #a2cffd; BORDER-BOTTOM-STYLE: none
}
TABLE {
	BORDER-RIGHT: #ffffcc 0px; PADDING-RIGHT: 0px; BORDER-TOP: #ffffcc 0px; PADDING-LEFT: 0px;
PADDING-BOTTOM: 0px; MARGIN: 0px; VERTICAL-ALIGN: top; BORDER-LEFT: #ffffcc 0px; PADDING-TOP: 0px;
BORDER-BOTTOM: #ffffcc 0px
}
TD {
	BORDER-RIGHT: #ffffcc 0px; PADDING-RIGHT: 0px; BORDER-TOP: #ffffcc 0px; PADDING-LEFT: 0px;
PADDING-BOTTOM: 0px; MARGIN: 0px; VERTICAL-ALIGN: top; BORDER-LEFT: #ffffcc 0px; PADDING-TOP: 0px;
BORDER-BOTTOM: #ffffcc 0px
}
TR {
	BORDER-RIGHT: #ffffcc 0px; PADDING-RIGHT: 0px; BORDER-TOP: #ffffcc 0px; PADDING-LEFT: 0px;
PADDING-BOTTOM: 0px; MARGIN: 0px; VERTICAL-ALIGN: top; BORDER-LEFT: #ffffcc 0px; PADDING-TOP: 0px;
BORDER-BOTTOM: #ffffcc 0px
}
TH {
	BORDER-RIGHT: #ffffcc 0px; PADDING-RIGHT: 0px; BORDER-TOP: #ffffcc 0px; PADDING-LEFT: 0px;
PADDING-BOTTOM: 0px; MARGIN: 0px; VERTICAL-ALIGN: top; BORDER-LEFT: #ffffcc 0px; PADDING-TOP: 0px;
BORDER-BOTTOM: #ffffcc 0px
}
.sub {
	FONT-WEIGHT: normal; FONT-SIZE: 12px; MARGIN: 0px; VERTICAL-ALIGN: top; COLOR: #000000;
TEXT-ALIGN: left; TEXT-DECORATION: none
}
.head {
	FONT-WEIGHT: bold; FONT-SIZE: 12px; MARGIN: 0px; VERTICAL-ALIGN: top; COLOR: #000000; TEXT-
ALIGN: left; TEXT-DECORATION: none
}
.piclink {
	COLOR: #c62065
}
.picnoborder {
	COLOR: #ffffcc
}
.linknoborder {
	BORDER-RIGHT: #000000 0px; BORDER-TOP: #000000 0px; MARGIN: 0px; BORDER-LEFT: #000000 0px;
BORDER-BOTTOM: #000000 0px
}
.txt_brown {
	COLOR: #702e2e
}
.txt_purple {
	COLOR: #94003e
}
.txt_blue {
	COLOR: #006085
}
.txt_lightblue {
	COLOR: #0065b7
}
.txt_red {
	COLOR: #820000
}
.txt_gray {
	COLOR: #b1b1b1
}
```