

Baby Monitoring System (BMS)

By

Muhammad Fahmi Bin Rosli

Dissertation submitted in partial fulfillment of

the requirements for the

Bachelor of Technology (Hons)

(Business Information System)

JANUARY 2006

Universiti Teknologi PETRONAS

Bandar Seri Iskandar

31750 Tronoh

Perak Darul Ridzuan

t

QA

76.76

1194

mas 2

2006

1) XML (Document Markup Language)

2) IT IS -- thesis

CERTIFICATION OF APPROVAL

Baby Monitoring System

By

Muhammad Fahmi Bin Rosli

A dissertation submitted to the
Business Information Systems Programme
Universiti Teknologi PETRONAS
in partial fulfillment of the requirements for the
BACHELOR OF TECHNOLOGY (Hons)
(BUSINESS INFORMATION SYSTEMS)

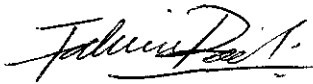
Approved by,

(Mr. Saipunidzam Mahamad)

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK
Semester January 2006

CERTIFICATE 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 have not been taken or done from unspecified sources or persons.



(MUHAMMAD FAHMI BIN ROSLI)

ABSTRACT

Baby Monitoring System is a system that being developed for helping the elderly in monitoring the activity of a toddler. Hence, the objective of this system is to help parents or the guardians in monitoring their baby in the baby's room whilst they are at another place within the residence. The scope of study for this system is being performed around the residence place in which that there are family with a baby and also whereby parents are having trouble in monitoring their baby. For developing this system, the author is using the Rapid Application Development (RAD) as for the methodology and also the environment concerning in developing such system. The methodology consists of four phases that is Requirements Planning, User Design, Construction and Cut Over. There are results obtain based on the studies made by the author during the development of the system. These results are being analyzed by the author in developing the system. The expected output from the results will be implementing by the author for the development of the system. As a conclusion, the system is being developed for helping the parents or guardians by using the feedback from the participants.

TABLE OF CONTENTS

CERTIFICATION	iii
ABSTRACT	iv
TABLE OF CONTENTS	v
LIST OF FIGURES	viii
LIST OF TABLES	ix
ABBREVIATIONS AND NOMENCLATURES	x
1.0 INTRODUCTION	1
1.1 Background of Study	1
1.2 Problem Statement	2
1.2.1 Problem Identification	2
1.2.2 Significance of the project	2
1.3 Objectives and Scope of Study	3
1.3.1 Objectives	3
1.3.2 Scope of Study	3
1.3.3 Relevancy of the Project	4
1.3.4 Feasibility of the Project within the Scope and Time Frame	4
2.0 LITERATURE REVIEW AND THEORY	5
2.1 Baby Monitoring System	5
2.2 Developing a Good and Reliable System	6
2.3 Video Streaming	6
2.4 Voice Recognition	7
2.5 Reminder System	7
2.6 Extensible Markup Language (XML)	8

3.0 METHODOLOGY	9
3.1 Procedure Identification	9
3.1.1 Preliminary Investigation	9
3.1.2 Analysis	10
3.1.3 Design	10
3.1.4 Construction	11
3.1.5 Implementation	12
3.2 Methods	12
3.3 Tools	14
3.3.1 Software	14
3.3.2 Hardware	15
4.0 RESULT AND DISCUSSION	16
4.1 Introduction	16
4.2 System Design	16
4.2.1 System Structure	16
4.2.2 The System Flow	19
4.3 Database Design and Database Structure	19
4.4 Distracted by infants	21
4.4.1 Baby Monitoring Techniques	22
4.4.2 System Spending Wise	23
4.4.3 Reminder System	25
4.5 Testing Phase & Results	27
4.6 Findings	28
4.7 Discussion	28
4.8 The Main Page Interface	29
5.0 CONCLUSION AND RECOMMENDATION	30
5.1 Conclusion	30
5.2 Recommendations	31

REFERENCES	32
APPENDICES	34
APPENDIX A	34
APPENDIX B	37
APPENDIX C	40
APPENDIX D	42
APPENDIX E	44
APPENDIX F	47
APPENDIX G	50
APPENDIX H	60

LIST OF FIGURES

- Figure 3.1.3 : The Main Page
- Figure 3.2 : Method Use for Each Task
- Figure 4.2.1 : The Main Page
- Figure 4.2.1.1 : The Reminder Page
- Figure 4.4 : Baby Distraction Analyses
- Figure 4.4.1 : Baby Monitoring Techniques
- Figure 4.4.2 : System Costs over RM 1000
- Figure 4.4.2.1 : A Cheaper System
- Figure 4.4.2.2 : Expected Price to Pay
- Figure 4.4.3 : Does Busyness Affect an Event
- Figure 4.4.3.1 : Medium Used As Reminder
- Figure 4.6 : Does the System Helps in the Parents/Guardians Daily Life

LIST OF TABLES

Table 2.1	How Much Does Your Baby Sleep
Table 3.2	Method Use for Each Task
Table 3.3.1	Software Requirement
Table 3.3.2	Hardware Requirement
Table 4.5	Testing Results

ABBREVIATIONS AND NOMENCLATURES

BMS : Baby Monitoring System
RAD : Rapid Application Development
VB : Microsoft Visual Basic
OOP : Object Oriented Programming
GUI : Graphical User Interface
SIDS : Sudden Infant Death Syndrome
CCTV : Close Circuit Television
PDA : Personal Digital Assistant
VB.Net: Microsoft Visual Basic.Net

CHAPTER 1

INTRODUCTION

1.1. Background of Study

Most of the parents or guardians have some problems in monitoring their toddler. This is because; parents have other activities to attend when they are at home other than monitoring the baby. However, it is also important for the parents to continuously monitor their baby on a regular basis. This is to ensure that there would be no harm occurs to the baby. For performing such activities, the parents will then have to perform multitasking at once in order to complete all the tasks.

In order for the parents to achieve these situations, the use of technology as a medium could be used for helping them in achieving such goals. Currently, there are many technology inventions that have eased the humans for monitor the activity of their toddler in their daily lives that are available in the market today.

Hence, after some research being made by the author, it is noted that not many parents know the existence of such system for ease their activities in monitoring the baby. The existing monitoring system that is available in the market is the traditional walkie-talkie where it uses two peripherals that connected through the radio frequency in which the parent could hear their baby's activity. Such old technology could be improved by implementing some functions to it. Hence, the Baby Monitoring System (BMS) uses modern technology that could provide better monitoring for the baby and also the ease of the parents.

1.2. Problem Statement

1.2.1. Problem Identification

Parents and guardians are having trouble in monitoring their baby from time to time. This is because; parents have to attend some other responsibility around the residence besides monitoring the baby. Such other responsibilities are office workload, house works and etc. With performing such responsibilities, there will be an issue when there is a toddler or a baby in the house. The reason being is because that as a normal human being, it is impossible for the parents to perform multitasking at a single period of time. With the lack of monitoring towards the baby, it could cause a hazardous situation where a baby could be injured or harm due to the actions.

Another problem that has being identified from the author is that, due to the hectic schedule of the parents' activities, it is potential for them to tend to forget some particular event. With events that are forgotten concerning the baby, it could bring some affects towards the baby's health and also their growth.

1.2.2 Significance of the project

The significance of the project is that to create a system that could help the parents in improving their baby in their daily lives. Through such system, it will capture the baby's activity and stream it through the video format and also voice format. This system allows the users to enter reminder(s) in which they tend to attend in the future. Through the development of the system, it is noted that it is feasible to be used in many situations such as the baby's room, in vehicles, hospital and etc.

Nonetheless, the author has made some studies in which that parents do not know the existence of the system and also does not use the system for helping them in monitoring their baby. Therefore, this will bring new approach in baby monitoring and also from a different angle of usage.

1.3. Objectives and Scope of Study

1.3.1. Objectives

Below are the objectives of the system:

1. To create a system that could help the parents in monitoring their babies in their daily lives.
2. To embed a voice activation/streaming system into a monitoring system developed.
3. To implement reminder function system in the system developed.

1.3.2. Scope of Study

For this project, the author will study on developing a monitor system and also a reminder system for the use of parents in monitoring their baby. The system will make ease for the parent in order for them to monitor their baby while being somewhere else to be doing other work. The author has decided to conduct the study around the residence of a family that has a new born baby or a toddler in the house. The reason of choosing such places, is because to have a closer research approach in gathering information for developing the system. Nonetheless, the study also will be conducted to particular groups of society such as middle wage earners and also high age earners. Hence, an effective and efficient monitoring system is useful to successfully fulfill their needs. In developing a reliable system, there are a number of technological and design requirements that should be considered hence, the author must consider and meet the requirements to ensure that it will ease the user's activity. Through the study, the author will also determine on how the Microsoft Visual Basic (VB) will incorporate with the external hardware to produce a dynamic system. The advantage of using VB compared to the other development tools is that it is easy to use and also uses Object Oriented Programming (OOP) which deals with real objects. The author also will have to study on how to develop such system that deals with real time programming in order for streaming purposes.

1.3.3 Relevancy of the Project

This project has its relevancy that is related to Information Technology. This is because the project is about introducing the use of system to parents in helping their daily lives. The development of the system involves the use of database using Microsoft Access. The author also has to apply all the knowledge in order to make such system a user friendly system. Thus many factors apply for such development such as graphical user interface (GUI), database, development programming and etc. Nonetheless, the author has made thorough studies about programming using Visual Basic.Net in which the author has some basics background of using the application. However, the author also has performed studies on how to set up a system that needs to be usable for all computer platforms that are currently available in the users' home. The user friendly interface is the key factor of the system in which the users of the system is from a beginner and also novice level of computer literates.

1.3.4 Feasibility of the Project within the Scope and Time Frame

In this project, the author is responsible to develop a stand alone system that will be use in monitoring babies. Through this system the users will have the ease in monitoring their baby while they are in some other places. The system also will have a reminder system in which it will help the users to track the events that is needed to be attended. For such system, it will be used by parents or guardians that have problems of monitoring their baby(s). The development process should be completed within four to five months.

CHAPTER 2

LITERATURE REVIEW AND THEORY

2.1 Baby Monitoring System

Baby monitoring system is basically being used in monitoring the baby whilst they are playing in they are playing in their play pan or in their rooms. Thus for such system it will be implement both at home and also in the hospital [1]. These monitors are intended for both home and clinical use. The baby monitoring systems monitor both the heart rate and respiration of infants to combat the sudden infant death syndrome (SIDS). Baby should have a constant sleep in which will help the growth of the toddler.

Table 2.1 How Much Does Your Baby Sleep

Age	Average	Range
0-1 months	16 ½ hours	10-22 hours
1-4 months	15 ½ hours	10-20 hours
5-8 months	14 ½ hours	9-20 hours
9-12 months	13 ½ hours	9-18 hours

The survey in Table 2.1 as discussed in [2] was taken from the guidance book. The table showed that the baby will decrease in average sleeping on a monthly basis. This is showed that as the baby growth, they are more tending to be awake when they are growing from time to time.

Nowadays there are also baby monitoring systems that are available in the market. Such systems are the Guardian Angel created by SWAN [3]. The system offers functions that are vary in monitoring purposes. However, these types of system are currently not available in the market and it is not widely used by the parents in helping them to monitor their toddler. It is recommended by other parents to use the baby monitoring system as a step of safetiness [4].

2.2 Developing a Good and Reliable System

“A good rule of thumb in design is that if the designer really likes some special feature, it should probably be the first thing to be discarded.” – Don Norman

When designing a system there are a number of technological and design requirements that should be considered [5]. According to Charlesworth Consulting article, it said that by considering and meeting the requirements will ensure that a good quality system is produced. There are a number of criteria that should be considered such as:

1. Availability
2. Scalability
3. Extensibility
4. Performance
5. Skills Availability
6. Future Proofing
7. Accessibility
8. Flexibility
9. Security

2.3 Video Streaming

Video Streaming is the use of technology in streaming videos through the use of medium peripherals in sending these types of multimedia. There are many uses of video streaming [6]. Such use of technology is as listed as per below:

- Training
- Meetings
- Conferences
- Monitoring
- Etc.

Thus also that video streaming could provides a continuous digital video and/or audio signal across a data network[7]. These types of technology need protocols to be followed in order for them to be workable in a propable manner. Firstly the users

need a personal computer (CPU) for compression, harddisk space for storage and also connected to a network for the distribution of the video streaming. There are also a stand alone software that are available for monitoring by using a single computer.

2.4 Voice Recognition

Voice recognition is the process of taking the spoken word as an input to a computer program [8]. Voice recognition is "the technology by which sounds, words or phrases spoken by humans are converted into electrical signals, and these signals are transformed into coding patterns to which meaning has been assigned".

The difficulty in using voice as an input to a computer simulation lies in the fundamental differences between human speech and the more traditional forms of computer input. While computer programs are commonly designed to produce a precise and well-defined response upon receiving the proper (and equally precise) input, the human voice and spoken words are anything but precise.

Each human voice is different, and identical words can have different meanings if spoken with different inflections or in different contexts. Several approaches have been tried, with varying degrees of success, to overcome these difficulties.

The voice recognition is being performed in three ways that is "template matching", "feature analysis" and also "speaker-independent". The template matching method of voice recognition is founded in the general principles of digital electronics and basic computer programming. There are many uses of of voice recognition. The examples of voice recognition are as per below:

- Automobiles [9]
- Hospitals [10]
- Games [11]
- Law Enforcement [12]
- Etc.

2.5 Reminder System

Reminder system is being use widely from time to time nowadays. This is because; with the workload and also the responsibilities needed for keeping track of meeting appointments or reminders that something needs to be done at a certain time [13]. There are many peripherals used by the users in order for implementing a reminder system. Examples of peripherals are such as hand phones, PDAs and etc. These reminder systems are customizable due to the needs of the users [14]. These reminder systems are being used in many purposes such as:

- Health issues
- Car Safety
- Appointments reminder
- Etc.

2.6 Extensible Markup Language (XML)

Extensible Markup Language (XML) is a simple, very flexible text format derived from Standard Generalized Markup Language (SGML) [15] . Originally designed to meet the challenges of large-scale electronic publishing, XML is also playing an increasingly important role in the exchange of a wide variety of data on the Web and elsewhere. The XML is a W3C-recommended general-purpose markup language for creating special-purpose markup languages, capable of describing many different kinds of data Nowadays, programmers are enforcing to use XML because of some factors. The factors are being listed as below:

1. With XML, the GUI is extracted. Thus, changes to display do not require futzing with the data. Instead, a separate style sheet will specify a table display or a list display.
2. Searching the data is easy and efficient. Search engines can simply parse the description-bearing tags rather than muddling in the data. Tags provide the search engines with the intelligence they lack.
3. Complex relationships like trees and inheritance can be communicated.
4. The code is much more legible to a person coming into the environment with no prior knowledge.

CHAPTER 3

METHODOLOGY

3.1. Procedure Identification

The methodology is basically the steps taken for the undergoing research and project work. In response to the faster pace of economy, Rapid Application Development (RAD) has become a popular route for accelerating system development [17]. As for this project, the author has decided to use the Rapid Application Development (RAD) techniques. The reason being is because the technique is also known as *spiral approach* where it emphasizes extensive user involvement in the rapid and evolutionary construction of working prototypes of the system in order to accelerate the system development process.

3.1.1 Preliminary Investigation

Based on the methodology chosen, the author has produced the project milestones to ensure that all the development stages are complete within the development stage as planned. Refer to Appendix A for the planned milestone.

Following are the tasks planned for each phase of the RAD technique for the development of the project. Refer to Appendix B for the RAD model.

During this phase, the author has performed some activities in order to complete this stage. The activities are being listed as per below:

1. Proposing the title of the project
2. Create the project scheduling
3. Analyze the problems of the project.

Hence, the author also has put into consideration on the behavior of the systems and also identifies the users and how they would use such system. The use case diagram of the system is being shown in the Appendix C.

3.1.2. Analysis

During this stage, the author has started to gather the information needed as required by the project. The requirement analysis is being performed to foresee what are the requirements needed for the project. There would be various techniques that will be applied in order for the information gathering to support the development of the system. Examples of information gathering are as per below:

1. Conduct Surveys
2. Distribute Questionnaires
3. Conduct Interviews

Examples of questionnaires are being place in Appendix D. Based on the feed back gathered from these information gatherings, the author will then conduct and also analyze the system requirements that will be use as the guidelines in developing the system. Hence, feasibility study is also being perform where the author will overview the feasibility of the project base on the scope and time frame.

3.1.3 Design

In the design phase, the author will design flow of the functionality of the system. This design flow shows the system operations and also the system's contents. Then the structure of the system will then take place where the plan will show the flow of the data. The interface of the system will also be designed during this phase. The system interface which is also known as Graphical User Interface (GUI) use as the presenter to the end users. The author will then conduct a survey in which the interface of the system would be designed. The example of the interface proposed by the author is as per below:

Figure 3.1.3 The main page

FILE	REMINDER	CLOCK
VIDEO STREAMING	LIST OF REMINDERS	
VIDEO STREAMING		

The figure above shows the propose interface by the author. The author will then conduct a survey on the user friendliness of the interface. The author also will conduct the usability of using various databases that is available in the market. With the result obtain, the author will then select the particular database that is suitable to be used in the system.

3.1.4 Construction

The construction phase will consume a great time for constructing the working prototype. The author will basically start work on the interface for the Baby Monitoring System (BMS) firstly. The system will be created by the use one of the *I-COMPUTER AIDED SOFTWARE ENGINEERING (I-CASE)* tools that is Microsoft Visual Basic (VB). By using the CASE tools, code generators, and prototyping tools provide a means of ensuring higher quality when employed using an appropriate methodology. The CASE environment allows the continuous in design. The end users are involving extensive close during this stage [17]. This is because; with the testing phase is being performed through out the construction phase in order to ensure that the prototype system will properly work. There are some steps that occurred during the construction stage. The steps are as per below:

1. Conduct survey to users
2. Design the system
3. Reviewed by users/conduct surveys

4. If approve, continue with the development, if not redesign until satisfy the user needs.

3.1.5 Implementation

The implementation phase will be the last phase in developing the Baby Monitoring System (BMS). After some testing phases that had been performed in the *construction phase*, the system is finally being agreed upon its functionality and also the GUI. Hence, there also will so additional test that are being conduct during this phase namely User tests and Integration tests. The author will conduct the User tests are to ensure that the program would run efficiently as intended. The other test which is Integration test is intend to perform in ensuring that the system produce by the author work with other application as they are put together. In the case where the system does not satisfy the requirement, the development cycle will loop back to the previous phase that is *analysis*. The author also has made some additional research on how to improve the system and also maintain the system.

3.2 Methods

Table 3.2 shows the methods applied by the author for the development of BMS.

Table 3.2 Method Use for Each Task

Task	Method
Initial Investigation	<p><i>Questionnaires</i></p> <ul style="list-style-type: none"> ▪ The author will prepare a set of questionnaires that will be given to the respective people in order to get their feedback towards the proposed system. The objective of the questionnaires is to gather as much information on the requirement of the system. <p><i>Observations</i></p> <ul style="list-style-type: none"> ▪ Observation on the baby's behavior, environment, culture, habits and many more are

	<p>needed to be handled. The information from these tasks will help in analyzing the requirement and design the system architecture.</p>
<p>Problem Analysis and Requirement Analysis</p>	<p><i>Interview</i></p> <ul style="list-style-type: none"> ▪ The author will conduct several informal interview sessions with some parents in order to perform some analysis. From this technique, the author will obtain a quick and direct data from the needed source. Nonetheless, it will also help the author in clarify doubts and ensure more understanding towards the project.
<p>Project Scheduling</p>	<p><i>Gantt Chart</i></p> <ul style="list-style-type: none"> ▪ For the project development, the author has produce a Gantt chart in performing a properly development work and tasks allocated for the project based on the time line given. The Gantt chart was created using the software of Microsoft Project.
<p>Designing the System Flow</p>	<p><i>System Flow Diagram</i></p> <ul style="list-style-type: none"> ▪ In designing such system, the flow of the system should be determined. The diagram will describe how the system operates. The flow diagram of the system is depicted in the Appendix.
<p>Construct/Coding</p>	<p><i>Visual Basic</i></p> <ul style="list-style-type: none"> ▪ The author has decided to incorporate the Visual Basic technology for the development of the proposed system.

Table 3.3: Tools

In order to develop such project, selected are being used for ensuring that it will successfully complete. The tools that are suggested to be used in the development of the system are as per below:

3.3.1 Software

Table 3.2 shows the minimum software requirement of the computer for the development of BMS.

Table 3.3.1 Software Requirement

No.	Software	Description
1.	Microsoft® Visual Basic.NET	This software is being use for Scripting Programming Language, Interface Design and also programming such systems.
2.	Microsoft® Sound Recorder	This will be use for voice detection system. It will be useful during the real time running of the system
3.	Adobe Acrobat Reader V6.0	This software will be used for the literature review part where some of the files need to be associated with such software.
4.	Creative Web Cam Driver	Since the author is using Creative Web Cam, therefore, it is needed for the system to connect to the web camera by using the driver.
5.	Microsoft Access	This tool performs as the back end for the system where it will be used for storing data. All the tables and fields require for this system will be created here

3.3.2 Hardware

Table 3.3 shows the hardware used throughout the development of BMS.

Table 3.3.2 Hardware Requirement

<u>No</u>	<u>Device</u>	<u>Requirement</u>
1.	Operating System	Microsoft Windows XP
2.	Processor	Intel Pentium 4 2.4GHz
3.	Memory	512 MB of memory
4.	Disk Space	40GB of free space
5.	Other Peripherals	Screen, Keyboard, Mouse, Camera, Microphone

CHAPTER 4

RESULT AND DISCUSSION

4.1 Introduction

The Author is responsible to develop such system namely Baby Monitoring System (BMS) which is being focused to be used by the people that are not so Information Technology (IT) literate or novice user of a computer and also which has baby in their residence. The system is basically to monitor the baby while they are in different place which are not within their parents sight. In addition, this system is also being use as a reminder to give the user the activity of tracking their appointed activities in which they have entered into the system. Fundamentally, the BMS is developed to help and also to ease the manual monitoring that is being practiced by the parents traditionally. Hence, parents could not monitor their baby constantly because they have some other responsibilities that they have to attend. The Baby Monitoring System focuses on two main functions which are to monitor the baby in different places, while the parents are somewhere else and also to help the users to remind about their appointments.

4.2 System Design

The Baby Monitoring System (BMS) consists about 20 pages including the associated files. The main parts of the system are the main page, create a new reminder and also edit the reminder page. Thus also the system consists of video monitoring and also voice trigger alarm.

4.2.1 System Structure

For the user interface, the author has designed a simple and usable page for the used of the system's users. The page uses a very basic yet attractive appearance, colors, and

font and other properties. The reason for implementing these types of properties is that, the author is trying to implement the most user friendly interface for the use of even the most beginner level of computer literates. The used of Style Sheet for the user interface can saves a lot of time by avoiding duplication while developing such system. Besides, the appearance of the system is easy to maintain and can be simply changed in the future in only one place.

Figure 4.2.1: The Main Page

FILE	REMINDER	CLOCK
VIDEO STREAMING	LIST OF REMINDERS	
VIDEO STREAMING		

The figure above is the structure of the default page or the main page for the Baby Monitoring System (BMS). The users will be directed to this page upon their loading of the system. There are basically 4 sections altogether; VIDEO STREAMING, LIST OF REMINDERS, FILE, REMINDER and CLOCK. The Main page is where the users are going to be use for the primary purposes of the system. File, reminder and also clock is the top menu for the BMS where it is link to another function. The main page is used to display video streaming and also the reminder listings.

Figure 4.2.1.1: The Reminder Page

The image shows a web form for entering reminders. It contains the following elements:

- Subject:** A single-line text input field.
- Location:** A single-line text input field.
- Date:** A two-line text input field for entering the date.
- Time:** A single-line text input field for entering the time.
- Reminder:** A single-line text input field.
- Comments:** A large rectangular text area for entering additional information.
- Buttons:** Two buttons labeled "Delete" and "Save" located at the bottom right of the form.

Figure 4.2.1.1 is the reminder page for the BMS. The layout for the page is simple and easy to understand to the users of the system. Hence, it also satisfies the user requirement where they usually are interested for the interface that is uncomplicated and ease of usage. The function of this page is basically to enter reminders into the database.

Brief discussion about the user interface consists in BMS are as per follow:

- ***Main Page***

This is the first page in which the user will attend to when they launch the system. The 2 video streaming columns are for displaying the video captured from the web cam which is located on a different place. There would also be the reminder lists section in which it displays/lists all the reminders entered by the users. All the reminders are kept in the database of the BMS. The main page also consists of the File menu in which the users will select the camera selections and also the exit function. The reminder menu consists of creating the new reminder and also editing the existing reminder.

- ***Reminder page***

This section is one of part included in the BMS. As the purpose of developing the BMS is to help the parents in monitoring the babies, there for this type of functionality is crucial for such system. This type of functions helps the parents or the guardians to monitor and track down the event that is needed to be attended. The BMS gives the users an easiness of storing the data/reminder in a simple database in which they are provided with the user friendly interface/form on the creation of the reminder. The users will have to enter the subject of the event, location of the event, time of the event and also the reminders in which how long does the users intend the reminder to be acknowledge. The users could also give comments towards the reminder that they would like to enter. After providing with all the information needed, the users need to click the Save button in order for the reminder to be saved in the database. This page also gives the users the ability of updating the information of the data entered. For such event, the users need to click the edit reminder section from the reminder menu or from the lists that is located at the main page of the system. The reminder page also gives the option of deleting and also reviewing other reminders at once.

The author has made research and also given out samples to users in which they have agreed with the design of the interface for the system.

4.2.2 The System Flow

The flow of the system is represented in Appendix E. The flowchart showed the functionality of the system. The most important phase that can be seen through the diagram is the video streaming and also the voice activation process.

4.3 Database Design and Database Structure

After creating the important face of the system, the author has implemented the database that is required for the reminder function for the system. A database is a collection of interrelated files which the data captured by an information system then stored in files and databases. Since this is a stand alone system, therefore there are no client server structures needed for the design of the database.

As for the database structure, the author had gathered all the information needed from the questionnaires. From the information, the author has decided in which attributes is important to be included or included in the database. Thus the attributes that are being identified are as per below:

1. Reminder_ID (Primary Key)
2. Subject
3. Reminder_Location
4. Reminder_DateReminder
5. Reminder_DateCreate
6. Reminder_RemindTime
7. Reminder_DatePopOut
8. Reminder_Description
9. Reminder_Status

For each fields inside the tables, there must be have the data type as to identify what type of each fields will be. The examples of data type are such as VARCHAR, INTEGER, NUMBER, DATA and etc.

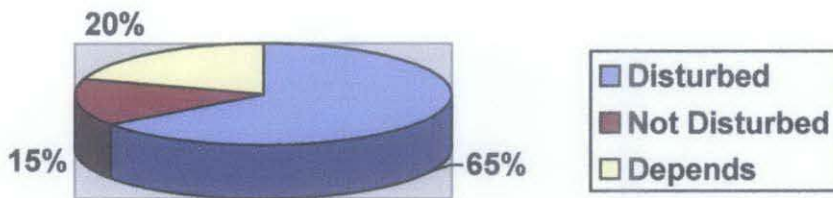
After all the gathered data is being segmented into their respective table, the author has identified the primary and also the foreign keys for all attributes since this system only uses one (1) tables because it is a standalone system. Thus, the database is then being tally with the system (computer) time in which it will be checking it with the Reminder_RemindTime, Reminder_DateReminder and also the Reminder_DatePopOut attributes. When all these attributes are being matched with the system's time are tally, therefore, the system will deploy a warning/reminder message to the users. The databse schema was manually created or designed on paper before it was being transferred into the application. The application that was being used in such system is the Microsoft Access. The reason that the author is using such application is because; most of the users that have or has a computer is using Microsoft programs.

The database, checks its lists regularly by the system in order to deploy the reminder as per setting by the users. The system checks its lists on an iterative basis. The system will check the time being set by the users and also compare it with the operating system's clock. If it is not tally with the operating system's time, it will check again until the reminder is tally with the list. The system will then deploy the reminder to the users as its purpose of functionality. The flow of the database is being shown in the Appendix E.

4.4 Distracted by infants

The main objective of such system is to help the guardians and also the parents in monitoring their baby while they are doing some other activities in other places. Questionnaires were given to the participants/ candidates by the author. Examples of the questionnaires are being shown in Appendix D. After some analysis about the distractions from the infants, the author has come out with a pie chart regarding the question. The pie chart is being shown as follow:

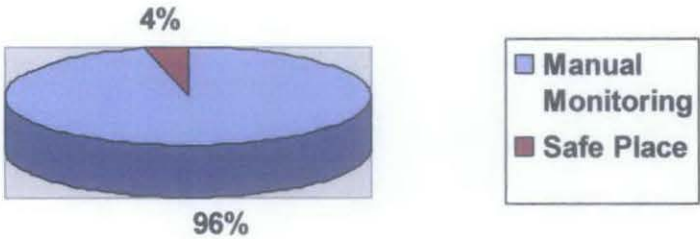
Figure 4.4 Baby Distraction Analyses



Based on the figure above, it shows the percentage of the results that are being answered by the participants of the questionnaires. The particular questions that are being asked by the author are the infants or the toddler that are being around the house or the residence; disturb the activities of the participants. Based on the figure above, it shows that sixty five percent (65%) of the participants agreed that having a child or the toddler does limit in performing their activities that they wish to intend to. The other fifteen percent (15%) and also twenty percent (20%) goes to do not disturb and also depends on the situations respectively.

4.4.1 Baby Monitoring Techniques

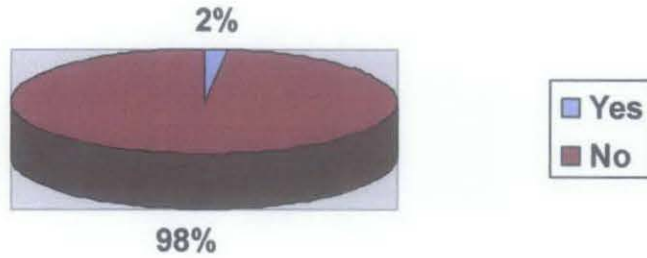
Figure 4.4.1 Baby Monitoring Techniques



Based on the figure above, it shows the response of the participants regarding the monitoring techniques that they are practicing in monitoring their baby or the toddler. The questions that were being to the participants are provided in the questionnaires being given by the Author. The particular question that was being answered in this segment and also regarding the figure given (Figure 4.2) is on how the parents or the guardian monitor their baby in their daily activities. It is being proven that most of the parents or the guardians monitor their baby manually in which says that ninety six percent (96%) of total participants practice such technique. The manual monitoring technique means that the baby is being placed within their (parents or the guardian) sight in which they could easily monitor. This method of monitoring do limit to the parents and also the guardians activity. The other four percent (4%) of the participants monitor their baby by putting them in a safe place where they will attend to their toddler when they here a sound from their toddler.

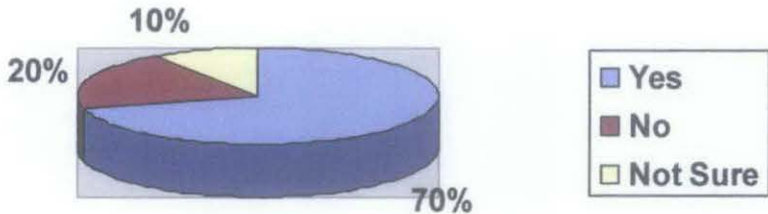
4.4.2 System Spending Wise

Figure 4.4.2 System Costs over RM 1000



The figure above (Figure 4.3) shows the response of the participants to the question whether they are willing to pay a system that costs over RM 1000 for only monitoring their baby. Thus, the results show that ninety eight percent (98%) of the participants will not buy such system and in which two percent (2%) will consider of buying the system. The reason being is because that most families in the country (Malaysia) have an average wage of five thousand ringgit (RM 5000).

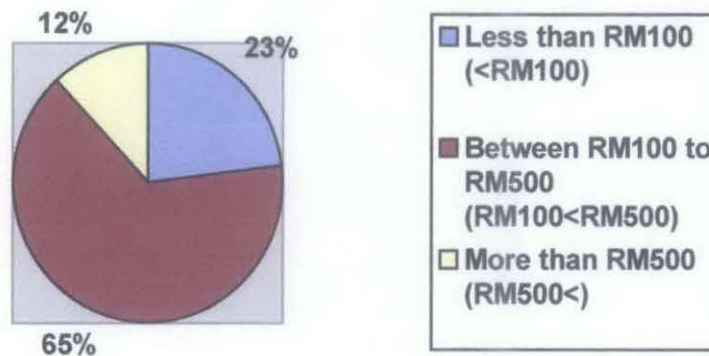
Figure 4.4.2.1 A cheaper system



When the questions regarding about the substitute system that would be cheaper compared to the CCTV, the results are being shown in the figure above (Figure 4.4). It is shown that seventy percent (70%) of the candidates would consider of buying such system when there are cheaper system available in the market. Twenty percent (20%) of the candidates would not consider of buying the system and also the remaining ten percent (10%) of the candidate are having doubts of buying the system. The reason of the seventy percent (70%) of the candidates would consider of buying the system is that because, with the appropriate costs or price for the system, they are willing to spend a

little bit extra of buying the system. The other candidates who are not willing to buy the system a cheaper price is because, they (the candidates) are willing to monitor their toddler manually in which from their point of view will be convenient for them because they have practiced it for generations. The remaining ten percent (10%) of the participant have the doubt of buying such system at the lower price.

Figure 4.4.2.2 Expected Price to Pay



The chart above shows the response from the candidates regarding their willingness of paying for such system. Thus, after some analysis by the author, it shows that sixty five percent (65%) of the participant would have the spare of financial of spending for such system of between RM100 to RM200. Twenty three (23%) of the participant would pay less than RM100 for the system and the remaining twelve percent (12%) would pay more than RM500 for the system. Therefore, after some consideration, the author has projected that the average price or cost for such system would be RM200. This is because; it is the mean price for market the system. With the mean price, mass marketing could be implemented in which the system could satisfy most of the consumers. The target market for this system is basically users that are not so Information Technology (IT) literate and also has problems in monitoring their baby. Hence, below are the price segmented for the system:

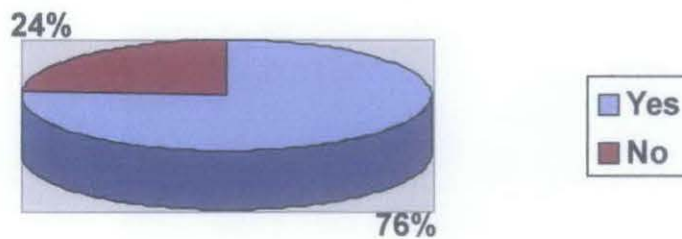
- *Web Cam – RM 100*
- *Microphone – RM 30*

- *System – RM 70*

Thus with the projected price being set, it shows that the developer or the author would have the payback of RM70 per system.

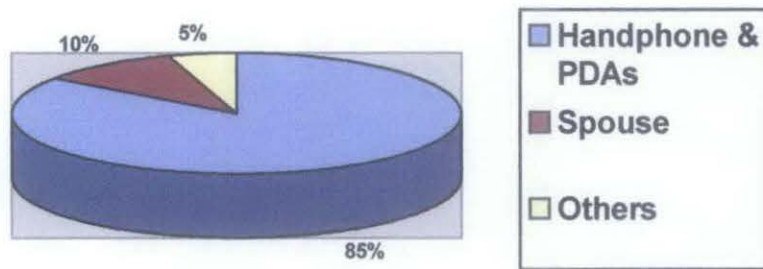
4.4.3 Reminder system

Figure 4.4.3 Does busyness affects an event



The figure above (Figure 4.6) shows the response from the participants regarding the questions of the effectiveness of busyness towards their daily life of an event. Thus, from the figure given, the author has analyzed that seventy six percent (76%) of the participants agrees that the busyness in their daily life does affect some of their daily activities such as appointments. This means that the busyness that the candidates are facing, sometimes make them forget things such as events that they should have attend such as meetings, medical appointments and etc. The other remaining twenty four percent (24%) of the candidates denies that the busyness in their daily life does not affect the event that they should attend.

Figure 4.4.3.1 Medium used as reminder



The figure above (Figure 4.7) shows the medium that is being used by the participants as a reminder for their events. After some analysis from the feedback from the participants, the Author has noted that eighty five percent (85%) of the participants uses their handphones or PDAs as their medium of reminder of an event. Some of the participants use their spouse as a medium of an event reminder in which the spouse will remind the other party (husband) of the event. This accumulates of fifteen percent (15%) from the overall analysis. Hence, the remaining five percent (5%) of the participants uses other techniques for reminding the event that they have to attend. Such technique that these participants use other than the other two are such as the secretary or even the traditional diary. After some analysis being made, the author has noted that the trend being practiced nowadays is to place a reminder inside the mobile gadgets such as handphones and also PDAs.

4.5 Testing Phases & Results

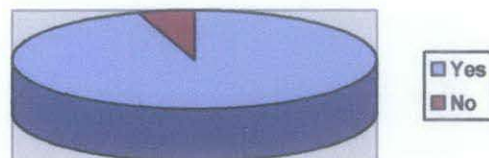
There were testing being made by the participants on the usability and also the functionality of the system. A total of 20 participants were invited to give the system a run for the purpose of achieving the objectives of the system. The results are being shown in the table below:

Table 4.5: Testing Results

Question	Objectives	Results
Camera Selection Menu	To select the cameras both for the primary and also the secondary camera	Both cameras are successfully selected from the lists of selection given
Camera Switch Button	The button for switching from the primary camera to the secondary and vice versa	The button successfully its function
New Reminder Menu	To enter a new reminder into the system	Reminders are successfully entered into the system with the ease of use
Edit Reminder Menu	To edit and also delete the reminders that are available in the lists	All reminders can be edited and also deleted from the lists
Reminder Time Selection	The deploy of the reminder time base on the user's selection	All users' selection for deploying the reminder are successfully deploy
Reminder Self Checking	The system checks its reminder list frequently for deploying the reminder	The system successfully checks the reminder lists for deploying the reminder.

4.6 Findings

Figure 4.6: Does the system helps in the parent's/guardians daily life



The figure above shows the perception on the users regarding of the system's feasibility in helping them in their daily life or vice versa. It shows that 95% of the users agrees that the system will make ease of their activities in monitoring the toddler or baby and the other 5% think that the system is not something that is important. The numbers of participant in the questionnaire and the indirect interview was 30 people altogether. Base on the findings, it is proven that the system really would help the parents or the guardians in monitoring the toddler or the baby.

The system is being analyzed and also reviews in order to fulfill the needs and wants of the users. The author also has made the system as user friendly as possible for the purpose of ensuring that they system will help the users that have the computer skills of beginners and also novice.

4.7 Discussion

From the result obtained by the questionnaire, the author could conclude that parents or guardians are having trouble in monitoring their toddlers or babies. Most of the participants are looking forward for such system to be available in the market whereby it is not available yet in the country. The participants believe that with such systems being implemented in their daily life, it could ease them in performing their daily activity whilst monitoring their baby from time to time.

4.8 The main page interface

The interface of main page of the system is being presented in the Appendix F. The manuals are also included in the appendix.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

In conclusion, the system is focusing on the monitoring baby's monitoring which will be used to help the parents or guardian in their daily live. The system is to make the functionality of the system reliable to the users. The function consist in the Baby Monitoring System (BMS) are to monitor the baby with ease so that the users will have ease in doing other work. Thus, the system also has the functionality of reminding the users who are tend to forget any functions that they are supposedly to attend. This will give the users the advantage of tracking their activities. The good choice of the technology cooperated with the system is one of the factor that lead to a successful of the system development.

Successful development of Baby Monitoring System (BMS) will eliminate the difficulties and problems face by the parents or the guardian of a baby. The proposed system is much reliable compared to the traditional or available system in the markets in terms of its functionality and also the reminder system. Furthermore, the system can improve the quality and also the functionality of the product by enhancing multiple functions to it.

The implementation of such project gives a good exposure to the author as the project required output of working model. This gives opportunities to the author in discovering new management skills, knowledge and applying the knowledge gained through working life. Good project planning and diligent work is very important along with supervision from the supervisor in charge. These are among the key factors that can lead to the success of the project.

5.2 Recommendations

Although this project is completed, there are several recommendations that can be applied in respect for the future development.

Firstly, the author would like to suggest that the system would have motion detection. In this project, the author is only using the web cam for monitoring the baby. Thus, with the use of motion detection, the web cam or the system will capture the image and if there are any motions being made by the toddler in whom the system will trigger an alarm to the parents or the guardians.

Secondly, the author would suggest or recommend that the system would have a wireless technology to be implemented to it. For the current system, the alarm is being acknowledged at the user's computer in which it is being placed at a stationery place. Thus, with implementing the wireless technology, the use of a handheld device as a receiver of the alarm, would be a step ahead in making it convenient to the users. Nonetheless, the system also would be able to use a wireless camera in which no wires are needed for connecting to the system.

Lastly, the author would suggest a voice activation system. Such technology would be a convenient for the users to use the system. The input of the reminders and also the monitoring session would be much simpler if it is being made or entered into the system by using verbally instead of manually typing it to the system. Other than that, the author suggests that the interface of the system to be more attractive.

REFERENCES

[1] FUSE Case Study (Ref.No.: 24571),

www.fuse-network.com/fuse/demonstration/331/24571/index.html

[2] Signe Larson, Kevin Osborn (1999), "*The Complete Idiot's Guide to Bringing Up Baby*", alpha books

[3] SWAN, Guardian Angel,

http://www.swann.com.au/new%20brochures/Security%20Range/Guardian%20Angel/Wireless%20GA_SW-G-GAW.pdf

[4] Reader's Comment, 2006, "*Should we use a baby monitoring system,*",

<http://babycenter.com>

[5] Charlesworth Consulting, "*What Makes a Good System*",

http://web.onetel.net.uk/~mj_charlesworth/whatmakesagoodsystem.htm

[6] O'Brien, Lyons, Hourigan,

<http://www.heanet.ie/conferences/2004/presentations/thursday/Multimedia%20Services%20for%20conference.pdf>

[7] Jaromil, http://korova.dyne.org/video_streaming.pdf

[8] Jim Baumann,

<http://www.hitl.washington.edu/scivw/EVE/I.D.2.d.VoiceRecognition.html>

[9] BMW USA, http://www.bmwusa.com/NR/rdonlyres/F98EF5A3-D392-4252-A282-DF3AEAA9178A/0/2006_e65_Broch.pdf

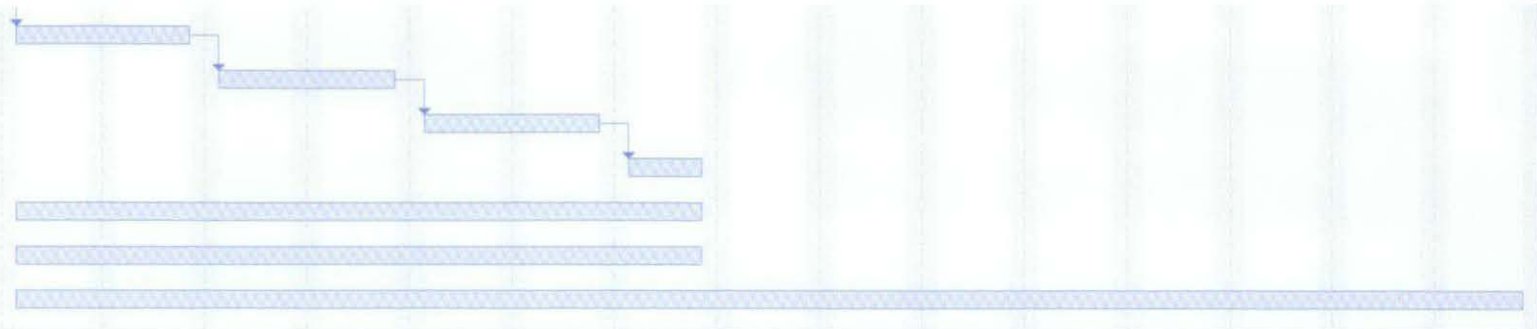
[10] Voice Recognition System, 2006, <http://talktoyourcomputer.com/about.html>

- [11] Tiger Electronics, 1998, "*Giga Pets Plus*",
<http://www.hasbro.com/common/instruct/70-207.PDF>
- [12] Mark P. Cohen, 2003, "*The Voice Response Translator: A Valuable Police Tool*",
- [13] LookSmart, 2006, "*Creating an appointment system*",
<http://looksmartsolutions.com>
- [14] Integrated Office Systems Inc., 2003, "*Integrated Office System Corporation*",
<http://iosc.com>
- [15] W3C, 2006, "*Extensible Markup Language (XML)*", <http://www.w3.org/XML/>
- [16] Adam Kolawa, 2004, "*The ABCs of XP, RAD, and PSP*",
<http://www.stickyminds.com/sitewide.asp.htm>
- [17] Jeffrey L. Whitten, 2001, "*System Analysis and Design Methods, 5th Edition*",
McGraw Hill

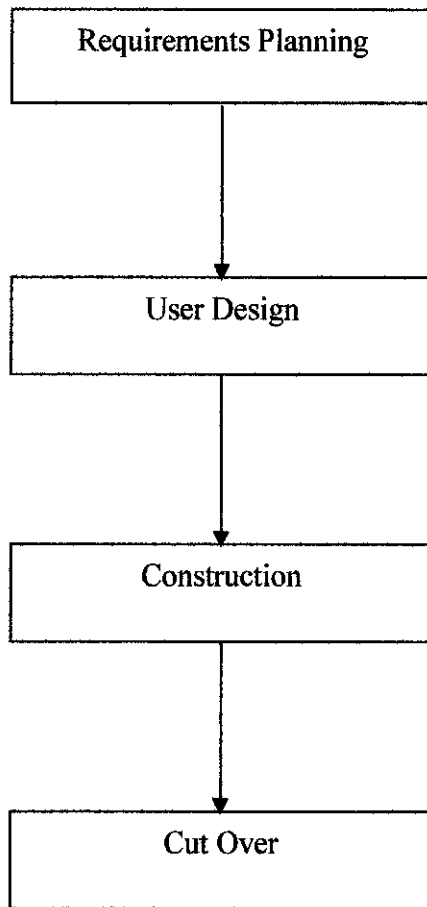
APPENDICES

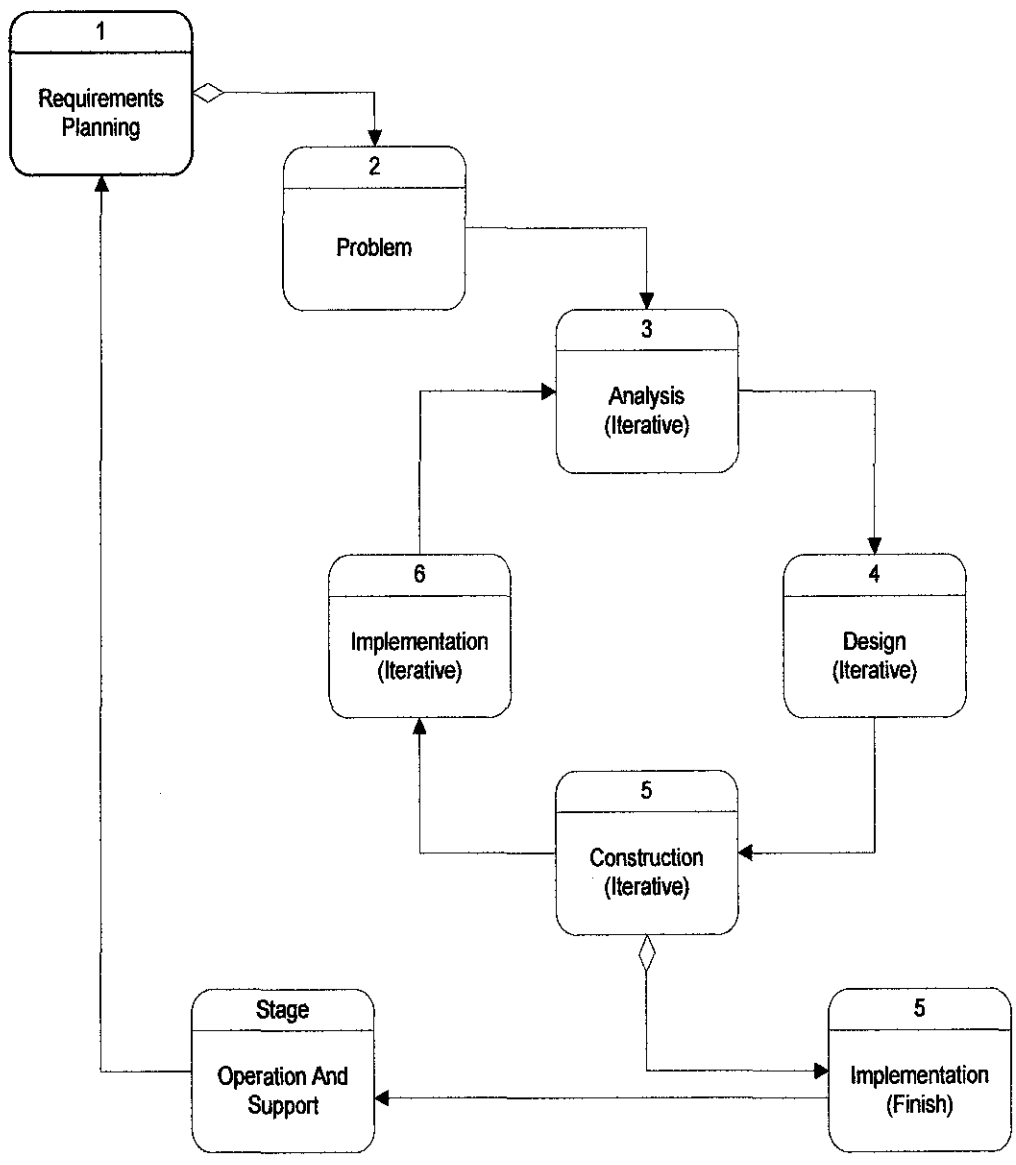
APPENDIX A:
Project Timeline

2	Requirements Planning	10 days?	Mon 30/01/06
3	User Design	10 days?	Mon 13/02/06
4	Construction	10 days?	Mon 27/02/06
5	Cut Over	5 days?	Mon 13/03/06
6	Research	35 days?	Mon 30/01/06
7	Progress Report	35 days?	Mon 30/01/06
8	Final Report	75 days?	Mon 30/01/06



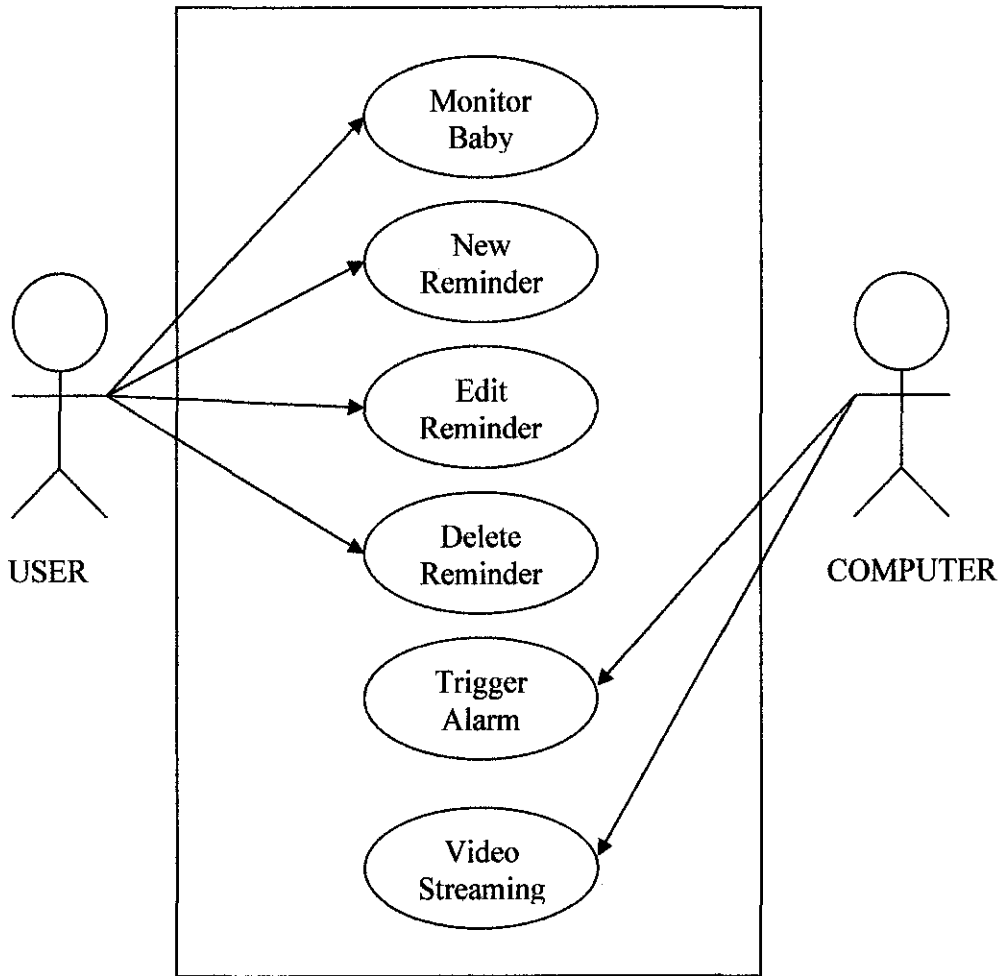
APPENDIX B:
Rapid Application Development (RAD) Model





**APPENDIX C:
USE CASE DIAGRAM**

BABY MONITORING SYSTEM



APPENDIX D:
Questionnaires

Baby Monitoring System (Questionnaire)

1. Where do you place your baby when they are sleeping?

2. How do you monitor your baby (When you have something else to do):

—

3. When you have some activities to do around the house, do you carry/ wake up the baby when they are sleeping? *Yes/No/Depends*

4. Does your style of monitoring the baby sometimes burden or limit your activity?
Yes/No/Not Sure

5. Would you spend thousands of ringgit (RM) to just monitor your toddler/baby?
Yes/No

6. If there is a cheaper monitoring system (technology) offered by the market, would you consider of buying it? *Yes/No/Not Sure*

7. How much would you willing to pay for such system? RM _____

8.

9. Does busyness is one of the factors that you might forget an event that you should attend? *Yes/No/Not Sure*

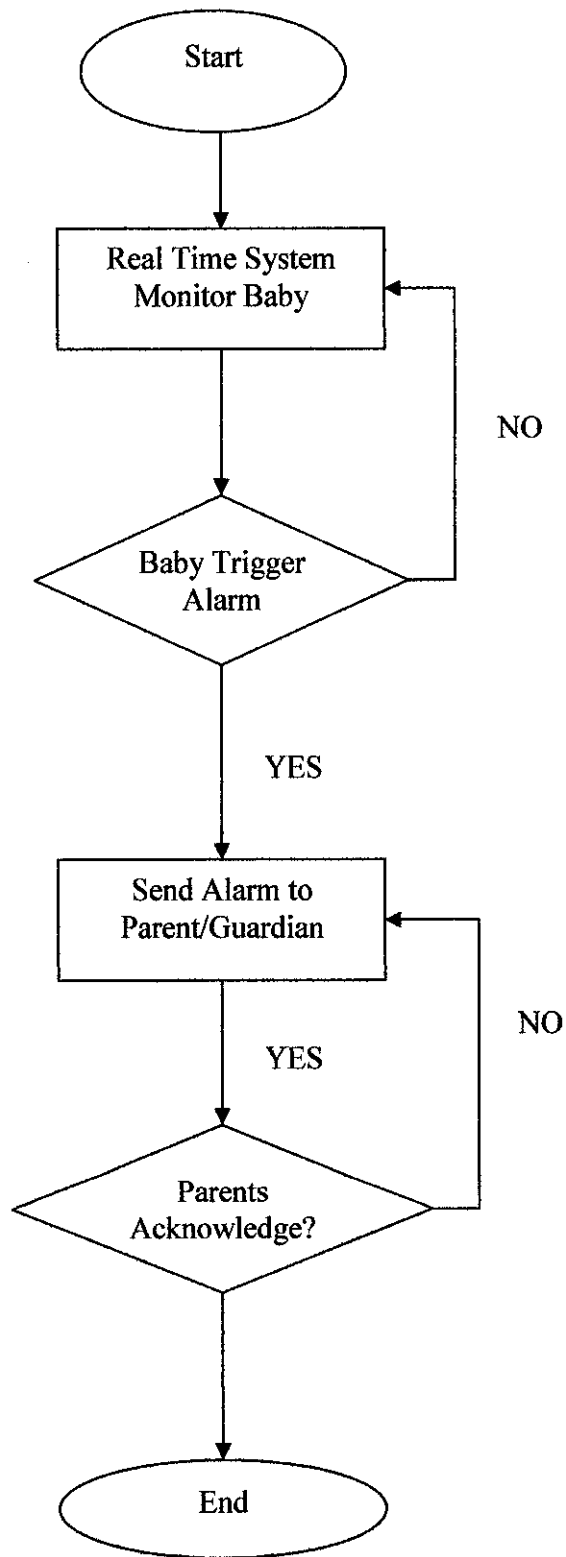
10. How do you overcome this problem (question 7)?

—

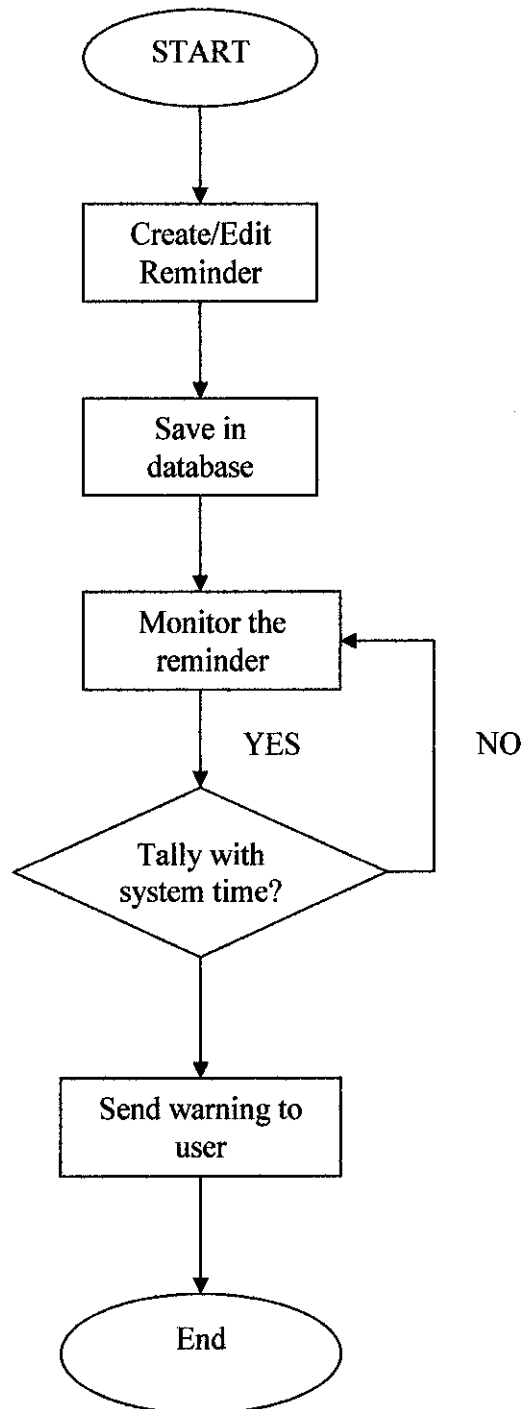
APPENDIX E:

BMS System Flow

BABY MONITORING FLOW



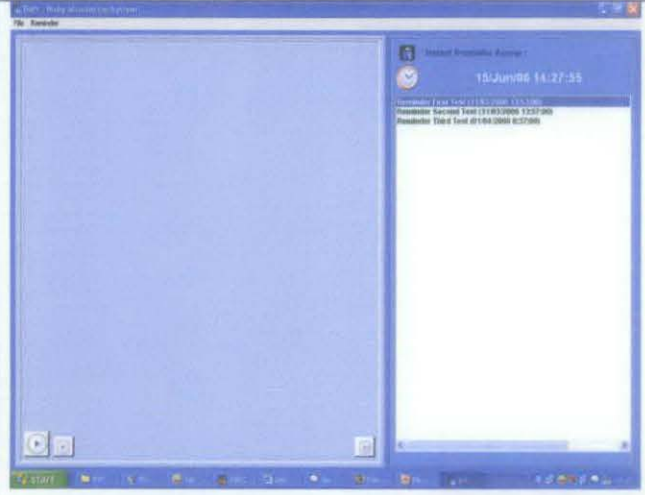
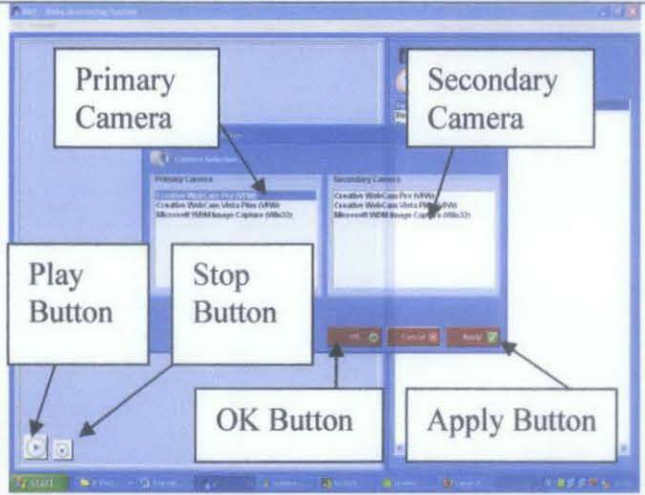
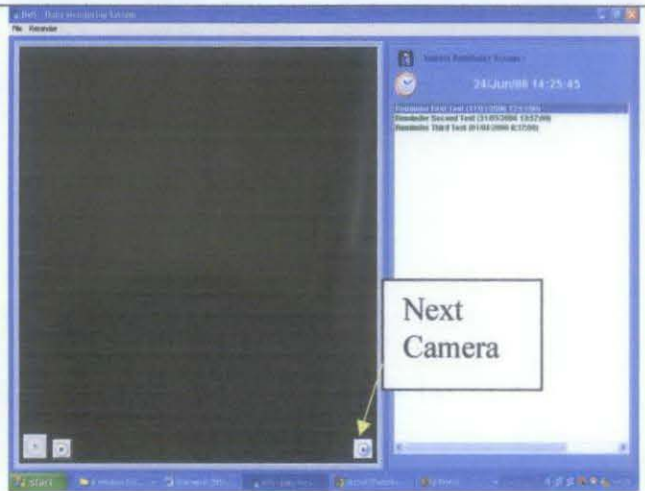
REMINDER SYSTEM FLOW



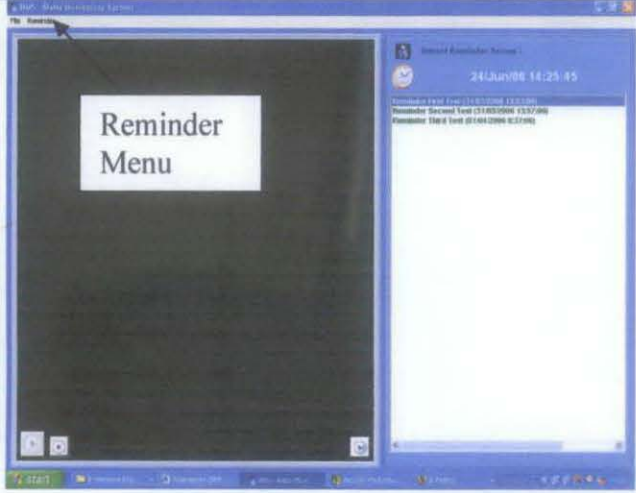
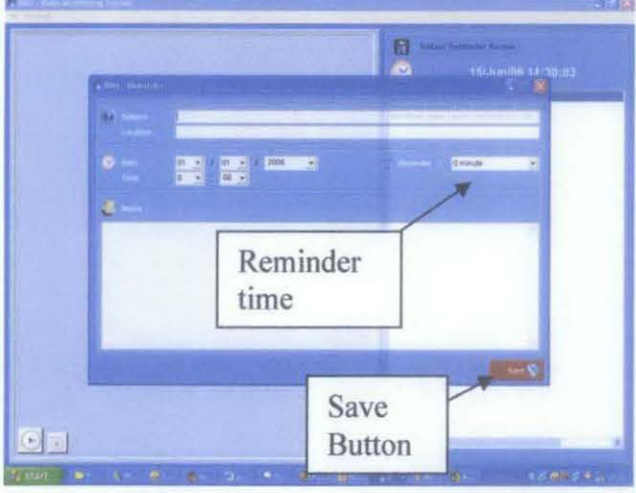
APPENDIX F:

BMS USER MANUAL

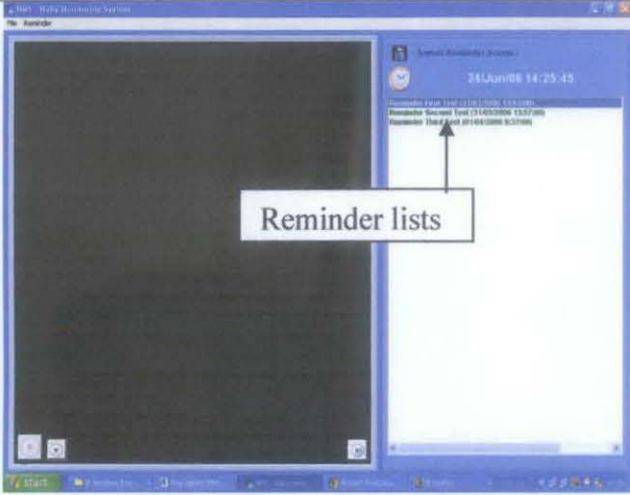
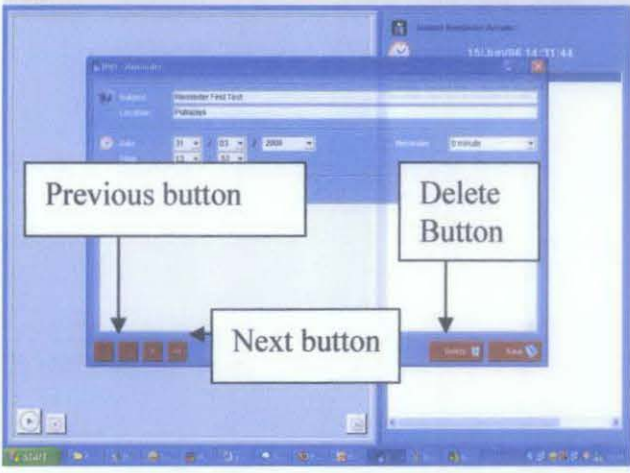
CAMERA SETTING & MANUAL

INTERFACE	INSTRUCTION
	<ol style="list-style-type: none"> 1. Select the file menu 2. Click on Camera Settings
	<ol style="list-style-type: none"> 1. Select the primary camera and the secondary camera 2. Click apply then click OK 3. Click the play button to start the camera, and stop button to stop the camera
	<ol style="list-style-type: none"> 1. The screen will alternately change from each camera to the other every 5 seconds. 2. To change to the other camera manually, simply click the next camera button.

New Reminder

Interface	Instruction
	<ol style="list-style-type: none">1. Go to the reminder menu2. Click New Reminder
	<ol style="list-style-type: none">1. Insert the particulars needed into the respective boxes2. Set when the reminder times before the event occurred.3. Click Save

Edit/Delete Reminder

Interface	Instruction
 <p>The screenshot shows the 'Reminders' application window. On the right side, there is a list of reminders. A white callout box with the text 'Reminder lists' has an arrow pointing to the list area. The list contains two items: 'Reminder: School Exam (21/03/2008 03:07:00)' and 'Reminder: School Exam (21/03/2008 03:07:00)'. The application title bar reads 'Reminders' and the system tray shows the date '26 Jun 08 14:25:45'.</p>	<ol style="list-style-type: none">1. Choose the reminder event that needs to be amend from the reminder list or the reminder menu (choose edit reminder)
 <p>The screenshot shows the 'Reminders' application window in edit mode. The window title is 'Reminders - Edit'. The form contains fields for 'Subject', 'Location', 'Date', 'Time', and 'Reminder'. Below the form, there are three buttons: 'Previous', 'Delete', and 'Next'. White callout boxes with arrows point to each of these buttons. The 'Delete Button' callout is positioned above the 'Delete' button, and the 'Next button' callout is positioned below the 'Next' button. The system tray shows the date '26 Jun 08 14:21:44'.</p>	<ol style="list-style-type: none">1. Amend the particulars in the boxes and also the reminder trigger2. To go to the next reminder click the next button or the previous button3. To save click the save button4. To delete click the delete button

**APPENDIX G:
BMS CODING**

```

'For interaction with ViewCam
Imports System.Runtime.InteropServices
'For XML
Imports System.XML

Public Class frmMain
    Inherits System.Windows.Forms.Form

    'Structure to store camera selection
    Public Structure CameraSettings
        Dim PrimaryCameraSelection As String
        Dim PrimaryCameraIndex As String
        Dim SecondaryCameraSelection As String
        Dim SecondaryCameraIndex As String
    End Structure

    'Variable for CameraSettings structure
    Dim udtCameraSettings As CameraSettings

    'Variable to keep track on how many reminder available in database
    Dim totalRows As Integer
    'Array to store reminder ID hiddenly (tag)
    Dim reminderID() As Integer
    Dim CurrentCamera As String

    'Declaring frmSettings object and also withevents to enable
    raising the event
    Dim WithEvents frmCameraSettings As New frmCameraSettings
    'Declaring frmReminder object and also withevents to enable
    raising the event
        Dim WithEvents frmReminder As New frmReminder

Const WM_CAP As Short = &H400S

    Const WM_CAP_DRIVER_CONNECT As Integer = WM_CAP + 10
    Const WM_CAP_DRIVER_DISCONNECT As Integer = WM_CAP + 11
    Const WM_CAP_EDIT_COPY As Integer = WM_CAP + 30

    Const WM_CAP_SET_PREVIEW As Integer = WM_CAP + 50
    Const WM_CAP_SET_PREVIEWRATE As Integer = WM_CAP + 52
    Const WM_CAP_SET_SCALE As Integer = WM_CAP + 53
    Const WS_CHILD As Integer = &H40000000
    Const WS_VISIBLE As Integer = &H10000000
    Const SWP_NOMOVE As Short = &H2S
    Const SWP_NOSIZE As Short = 1
    Const SWP_NOZORDER As Short = &H4S
    Const HWND_BOTTOM As Short = 1

    Dim iDevice As Integer = 0 ' Current device ID
    Dim hWnd As Integer ' Handle to preview window

    Declare Function SendMessage Lib "user32" Alias "SendMessageA" _
        (ByVal hwnd As Integer, ByVal wMsg As Integer, ByVal wParam As
Integer, _
        <MarshalAs(UnmanagedType.AsAny)> ByVal lParam As Object) As
Integer

```

```

    Declare Function SetWindowPos Lib "user32" Alias "SetWindowPos"
    (ByVal hwnd As Integer, _
        ByVal hWndInsertAfter As Integer, ByVal x As Integer, ByVal y
    As Integer, _
        ByVal cx As Integer, ByVal cy As Integer, ByVal wFlags As
    Integer) As Integer

```

```

    Declare Function DestroyWindow Lib "user32" (ByVal hwnd As
    Integer) As Boolean

```

```

    Declare Function capCreateCaptureWindowA Lib "avicap32.dll" _
    (ByVal lpszWindowName As String, ByVal dwStyle As Integer, _
        ByVal x As Integer, ByVal y As Integer, ByVal nWidth As
    Integer, _
        ByVal nHeight As Short, ByVal hWndParent As Integer, _
        ByVal nID As Integer) As Integer

```

```

    Declare Function capGetDriverDescriptionA Lib "avicap32.dll"
    (ByVal wDriver As Short, _
        ByVal lpszName As String, ByVal cbName As Integer, ByVal
    lpszVer As String, _
        ByVal cbVer As Integer) As Boolean

```

```

    Public Sub ReadSettings()
        'This sub is to interpret Settings.xml files
        'Then store the value of the elements inside a variable

        Try
            'Create new XmlTextReader object
            'Point to the file directory
            Dim textReader As XmlTextReader = New
    XmlTextReader("Settings.xml")
            'Set the whitespacehandling
            textReader.WhitespaceHandling = WhitespaceHandling.None

            ' Read until end of file
            Do While (textReader.Read())
                'Read the root tree (Settings)
                textReader.ReadStartElement("Settings")

                'Read the next node (CameraSettings)
                textReader.ReadStartElement("CameraSettings")

                'Read the PrimaryCamera node
                textReader.ReadStartElement("PrimaryCamera")

                'For CameraSelection (Primary)
                textReader.ReadStartElement("CameraSelections")
                udtCameraSettings.PrimaryCameraSelection =
    textReader.ReadString()
                textReader.ReadEndElement()

                'For CameraIndex (Primary)
                textReader.ReadStartElement("CameraIndex")
                udtCameraSettings.PrimaryCameraIndex =
    textReader.ReadString()
            End Do
        End Try
    End Sub

```

```

        textReader.ReadEndElement()

        'End the PrimaryCamera node
        textReader.ReadEndElement()

        'Read the SecondaryCamera node
        textReader.ReadStartElement("SecondaryCamera")

        'For CameraSelection (Secondary)
        textReader.ReadStartElement("CameraSelections")
        udtCameraSettings.SecondaryCameraSelection =
textReader.ReadString()
        textReader.ReadEndElement()

        'For CameraIndex (Secondary)
        textReader.ReadStartElement("CameraIndex")
        udtCameraSettings.SecondaryCameraIndex =
textReader.ReadString()
        textReader.ReadEndElement()

        'End the SecondaryCamera node
        textReader.ReadEndElement()

        'End the CameraSettings node
        textReader.ReadEndElement()

        'End the root tree(Settings) level
        textReader.ReadEndElement()
    Loop

    Catch ex As Exception
        ' Write exception message
        MsgBox("Error in reading Settings.xml. Please check this
file exists and giving the correct value", MsgBoxStyle.Exclamation Or
MsgBoxStyle.OKOnly, ex.Message)
    End
    End Try
End Sub

Public Sub LoadReminderListBox()
    'Clear the listbox first
    lstReminderHeader.Items.Clear()

    'Refresh the dataset
    DsBMS.Clear()
    totalRows = odaBMS.Fill(DsBMS, "tblReminder")
    ReDim reminderID(totalRows)

    Dim dr As DataRow
    Dim Counter As Integer = 0
    For Each dr In DsBMS.Tables("tblReminder").Rows
        'Populate the list box
        lstReminderHeader.Items.Add(dr("Subject"))
        'Get tag for hidden values
        reminderID(Counter) = dr("ID")
        'Increase Counter by 1
        Counter += 1
    Next

```



```

Next

If totalRows > 0 Then
    'Select Index 0 first
    lstReminderHeader.SetSelected(0, True)
End If
End Sub

Public Sub CheckReminder()
    'Refresh the dataset
    DsBMS.Clear()
    totalRows = odaBMS.Fill(DsBMS, "tblReminder")

    'Browse through each record
    Dim dr As DataRow
    For Each dr In DsBMS.Tables("tblReminder").Rows
        'Compare pop out date with current date n time
        If DateTime.Compare(Now, CDate(dr("DatePopOut"))) >= 0
Then
            'Check whether status = P (Pending)
            If dr("Status") = "P" Then
                'Pass the subject and datereminder
                modBMS.ReminderSubject = dr("Subject")
                modBMS.ReminderDateTime = dr("DateReminder")

                'Set status = C (Completed)
                dr("Status") = "C"

                'Beep
                Beep()
                'Show the reminder
                Dim frmPopOut As frmPopOut
                frmPopOut = New frmPopOut
                frmPopOut.Show()
            End If
        End If
    Next

    'Update dataset and commit changes
    odaBMS.Update(DsBMS)
    DsBMS.AcceptChanges()
End Sub

Private Sub frmReminder_ReminderUpdated() Handles
frmReminder.ReminderUpdated
    'Sub to handle RaiseEvent in frmReminder
    'Load reminder list box back after user has change it
    Me.LoadReminderListBox()
End Sub

Private Sub frmCameraSettings_SettingHasChanged() Handles
frmCameraSettings.SettingHasChanged
    'Sub to handle RaiseEvent in frmCameraSettings
    'Load settings back after user has change it
    Me.ReadSettings()
    If udtCameraSettings.PrimaryCameraSelection = "None" Or
udtCameraSettings.PrimaryCameraIndex = "None" Then

```

```

        'Still disable btnStart and btnStop
        btnStart.Enabled = False
        btnStop.Enabled = False
        btnNext.Enabled = False
    Else
        'Check whether camera is running
        If btnStop.Enabled Then
            ClosePreviewWindow()
        End If
        'Enable btnStart and disable btnStop
        btnStart.Enabled = True
        btnStop.Enabled = False
        btnNext.Enabled = False
    End If
End Sub

Public Sub OpenPreviewWindow()
    Dim iHeight As Integer = picCapture.Height
    Dim iWidth As Integer = picCapture.Width

    ' Open Preview window in picturebox
    hWndd = capCreateCaptureWindowA(iDevice, WS_VISIBLE Or
WS_CHILD, 0, 0, 640, _
    480, picCapture.Handle.ToInt32, 0)

    ' Connect to device
    If SendMessage(hWndd, WM_CAP_DRIVER_CONNECT, iDevice, 0) Then
        'Set the preview scale
        SendMessage(hWndd, WM_CAP_SET_SCALE, True, 0)
        'Set the preview rate in milliseconds
        SendMessage(hWndd, WM_CAP_SET_PREVIEWRATE, 66, 0)
        'Start previewing the image from the camera
        SendMessage(hWndd, WM_CAP_SET_PREVIEW, True, 0)

        ' Resize window to fit in picturebox
        SetWindowPos(hWndd, HWND_BOTTOM, 0, 0, picCapture.Width,
picCapture.Height, _
            SWP_NOMOVE Or SWP_NOZORDER)

        btnStop.Enabled = True
        btnStart.Enabled = False
    Else
        ' Error connecting to device close window
        DestroyWindow(hWndd)
    End If
End Sub

Public Sub ClosePreviewWindow()
    ' Disconnect from device
    SendMessage(hWndd, WM_CAP_DRIVER_DISCONNECT, iDevice, 0)
    ' Close window
    DestroyWindow(hWndd)
End Sub

Private Sub frmMain_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
    'MsgBox(Now)

```

```

'Load settings from Settings.xml
ReadSettings()

'Check first if the PrimaryCameraSelection <> None
If udtCameraSettings.PrimaryCameraSelection = "None" Or
udtCameraSettings.PrimaryCameraIndex = "None" Then
    'Disable btnStart and btnStop
    btnStart.Enabled = False
    btnStop.Enabled = False
    btnNext.Enabled = False

    'Show frmCameraSettings for user to select camera
    frmCameraSettings.Show()
Else
    'Enable btnStart and btnStop
    btnStart.Enabled = True
    btnStop.Enabled = True
    btnNext.Enabled = True
    CurrentCamera = "Primary Camera"
End If
picCapture.SizeMode = PictureBoxSizeMode.StretchImage

'Run tmrTime
tmrTime.Enabled = True
tmrTime.Start()

'Run tmrReminder
tmrReminder.Enabled = True
tmrReminder.Start()

'Get the reminder records and display in the listbox
LoadReminderListBox()
End Sub

Private Sub frmMain_Closing(ByVal sender As Object, ByVal e As
System.ComponentModel.CancelEventArgs) Handles MyBase.Closing
    'Confirm once user want to exit
    Dim Ans As Integer
    Ans = MsgBox("Do you really want to exit?",
MsgBoxStyle.Question Or MsgBoxStyle.YesNo, "Exit Confirmation")
    If Ans = vbYes Then
        'If yes, then end the application
        'Check whether the camera is running
        If btnStop.Enabled Then
            ClosePreviewWindow()
        End If
    End
Else
    'If no, just handle the event
    e.Cancel = True
    Exit Sub
End If
End Sub

Private Sub btnStart_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnStart.Click
    If CurrentCamera = "Primary Camera" Then

```

```

        iDevice = Val(udtCameraSettings.PrimaryCameraIndex)
    ElseIf CurrentCamera = "Secondary Camera" Then
        iDevice = Val(udtCameraSettings.SecondaryCameraIndex)
    End If

    If udtCameraSettings.SecondaryCameraIndex <> "None" Then
        'Enable btnNext
        btnNext.Enabled = True
    End If

    'Call sub routine to run the camera
    OpenPreviewWindow()
End Sub

Private Sub btnStop_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnStop.Click
    'Call sub routine to handles closing connection with the
camera
    ClosePreviewWindow()
    'Enable btnStart and disable btnStop
    btnStart.Enabled = True
    btnStop.Enabled = False
    btnNext.Enabled = False
End Sub

Private Sub btnNext_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnNext.Click
    If CurrentCamera = "Primary Camera" Then
        CurrentCamera = "Secondary Camera"
    ElseIf CurrentCamera = "Secondary Camera" Then
        CurrentCamera = "Primary Camera"
    End If

    'Call sub routine to handles closing connection with the
camera
    ClosePreviewWindow()

    If CurrentCamera = "Primary Camera" Then
        iDevice = Val(udtCameraSettings.PrimaryCameraIndex)
    ElseIf CurrentCamera = "Secondary Camera" Then
        iDevice = Val(udtCameraSettings.SecondaryCameraIndex)
    End If

    'Call sub routine to run the camera
    OpenPreviewWindow()
End Sub

Private Sub tmrTime_Tick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles tmrTime.Tick
    'For every tick of the timer (1 sec.) update the date and time
    lblTime.Text = System.DateTime.Now
End Sub

Private Sub tmrReminder_Tick(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles tmrReminder.Tick
    'For every tick of the timer (1 min.) check the database for
reminder

```

```

        CheckReminder()
    End Sub

    Private Sub mnNewReminder_Click(ByVal sender As System.Object,
    ByVal e As System.EventArgs) Handles mnNewReminder.Click
        'Set Reminder type as new reminder
        modBMS.ReminderType = "New"

        'Show frmReminder
        frmReminder.Show()
    End Sub

    Private Sub mnEditReminder_Click(ByVal sender As System.Object,
    ByVal e As System.EventArgs) Handles mnEditReminder.Click
        'Set Reminder type as new reminder
        modBMS.ReminderType = "Edit"

        'Show frmReminder
        frmReminder.Show()
    End Sub

    Private Sub mnCamSettings_Click(ByVal sender As System.Object,
    ByVal e As System.EventArgs) Handles mnCamSettings.Click
        'Show frmCameraSettings
        frmCameraSettings.Show()
    End Sub

    Private Sub mnExit_Click(ByVal sender As System.Object, ByVal e As
    System.EventArgs) Handles mnExit.Click
        'Call close event
        Me.Close()
    End Sub

    Private Sub lstReminderHeader_DoubleClick(ByVal sender As
    System.Object, ByVal e As System.EventArgs) Handles
    lstReminderHeader.DoubleClick
        'Check whether there is item in the listbox
        If lstReminderHeader.Items.Count > 0 Then
            'There is some items there
            'Set ReminderType as edit special
            modBMS.ReminderType = "EditSpecial." &
            reminderID(lstReminderHeader.SelectedIndex)
            frmReminder.Show()
        End If
    End Sub

End Class

```

**APPENDIX H:
DATABASE CODING**

```

Public Class frmReminder
    Inherits System.Windows.Forms.Form

    'Variable to track down what is the largest current ID in
tblReminder
    Dim largestID As Integer
    'Variable to track down what is the smallest ID in tblReminder
    Dim smallestID As Integer
    'Variable to track down what is the current ID that current being
used
    Dim currentID As Integer
    'Variable to track down how many rows are available in the
tblReminder
    Dim totalRows As Integer

    'Variables for when the pop up time and date should occurs
    Dim strPopOutFullDate As String
    Dim ReloadFlag As Boolean

    'Event whenever Reminder has been modified
        Public Event ReminderUpdated()

Public Sub ReloadForm()
    'Set largestID and currentID = 0 first
    largestID = 0
    currentID = 0
    'Set smallestID = 1 first
    smallestID = 1

    'First check what type of items need to be displayed
    If modBMS.ReminderType = "New" Then
        'Hide btnFirst, btnPrevious, btnNext, btnLast and
btnDelete
        btnFirst.Visible = False
        btnPrevious.Visible = False
        btnNext.Visible = False
        btnLast.Visible = False
        btnDelete.Visible = False
        'Display btnSave
        btnSave.Enabled = True

        'Reset all the controls
        txtSubject.Text = ""
        txtLocation.Text = ""
        cbxDay.SelectedIndex = 0
        cbxMonth.SelectedIndex = 0
        cbxYear.SelectedIndex = 0
        cbxReminder.SelectedIndex = 0
        cbxHour.SelectedIndex = 0
        cbxMinute.SelectedIndex = 0
        txtDescription.Text = ""
    ElseIf modBMS.ReminderType = "Edit" Then
        'Show all
        btnFirst.Visible = True
        btnPrevious.Visible = True
        btnNext.Visible = True
        btnLast.Visible = True

```

```

        btnDelete.Visible = True

        'Need to display data in the database
        LoadReminder()
Else
    'Show all
    btnFirst.Visible = True
    btnPrevious.Visible = True
    btnNext.Visible = True
    btnLast.Visible = True
    btnDelete.Visible = True
    btnSave.Visible = True

    'It is EditSpecial type
    'We need to know what is the smallest ID in the database
    TraceSmallestID()
    'We need to know what is the largest ID in the database as
weel
    TraceLargestID()
    'Extract the ID
    currentID =
CInt(modBMS.ReminderType.Substring(modBMS.ReminderType.LastIndexOf(".")
) + 1))
    FilterUI()
    LoadData(currentID)
End If
End Sub

Public Sub TraceLargestID()
    'This sub purpose is used to trace the current largest ID
value exist inside BMS
    Dim dr As DataRow
    Dim FirstRecordFlag As Boolean = False
    Dim tempID As Integer
    tempID = 0

    'Refresh the dataset
    DsBMS.Clear()
    totalRows = odaBMS.Fill(DsBMS, "tblReminder")

    'Browse through each row inside the NG STATUS TRANSITIONS
table
    For Each dr In DsBMS.Tables("tblReminder").Rows
        If dr.Item("ID") <> 0 Then
            'Assign smallestID equal to the first record
            If FirstRecordFlag = False Then
                largestID = dr.Item("ID")
                'Set flag as true
                FirstRecordFlag = True
            End If
            'Assign tempID equal to the current row ID value
            tempID = dr.Item("ID")
        End If
        'If currentID less or equal the tempID then we replace its
value
        If largestID <= tempID Then
            largestID = tempID

```



```

        End If
    Next
End Sub

Public Sub TraceSmallestID()
    'This sub purpose is used to trace the current largest ID
value exist inside BMS
    Dim dr As DataRow
    Dim FirstRecordFlag As Boolean = False
    Dim tempID As Integer
    tempID = 0

    'Refresh the dataset
    DsBMS.Clear()
    totalRows = odaBMS.Fill(DsBMS, "tblReminder")

    'Browse through each row inside the NG STATUS TRANSITIONS
table
    For Each dr In DsBMS.Tables("tblReminder").Rows

        'Assign tempID equal to the current row ID value
        If dr.Item("ID") <> 0 Then
            'Assign smallestID equal to the first record
            If FirstRecordFlag = False Then
                smallestID = dr.Item("ID")
                'Set flag as true
                FirstRecordFlag = True
            End If
            tempID = dr.Item("ID")
        End If
        'If smallestID greater or equal the tempID then we replace
its value
        If smallestID >= tempID Then
            smallestID = tempID
        End If
    Next
End Sub

Public Sub FilterUI()
    If totalRows <= 1 Then
        If totalRows = 0 Then
            'Disable save button
            btnSave.Enabled = False
        Else
            'Enable save button
            btnSave.Enabled = True
        End If
        'Disable all navigation buttons
        btnFirst.Enabled = False
        btnPrevious.Enabled = False
        btnNext.Enabled = False
        btnLast.Enabled = False
        'Exit Sub
        Exit Sub
    End If
    'Sub to filter whether we should
    If currentID = smallestID Then

```

```

        'Disable btnFirst and btnPrevious
        btnFirst.Enabled = False
        btnPrevious.Enabled = False
        'Enable btnNext and btnLast
        btnNext.Enabled = True
        btnLast.Enabled = True
        'Enable btnSave
        btnSave.Enabled = True
    ElseIf currentID = largestID Then
        'Disable btnNext and btnLast
        btnNext.Enabled = False
        btnLast.Enabled = False
        'Enable btnPrevious and btnFirst
        btnFirst.Enabled = True
        btnPrevious.Enabled = True
        btnSave.Enabled = True
    Else
        'Enable all navigation buttons
        btnFirst.Enabled = True
        btnPrevious.Enabled = True
        btnNext.Enabled = True
        btnLast.Enabled = True
        btnSave.Enabled = True
    End If
End Sub

Public Sub CheckIDValidity(ByVal Type As String)
    'Refresh the dataset
    DsBMS.Clear()
    totalRows = odaBMS.Fill(DsBMS, "tblReminder")

    'Search data for the ID given
    Dim FoundRow() As DataRow
    FoundRow = DsBMS.Tables("tblReminder").Select("ID='" &
currentID & "'")
    If FoundRow.Length = 0 Then
        'There is no record, do something here
        If Type = "Increment" Then
            'Keep on increasing by 1
            currentID += 1
        ElseIf Type = "Decrement" Then
            'Keep on decreasing by 1
            currentID -= 1
        End If
        'Call back this sub routine
        CheckIDValidity(Type)
    Else
        'It is valid
        'Abort the sub
        Exit Sub
    End If
End Sub

Public Sub CalculatePopUpTime()
    'Check the remind time
    Dim PopOutFullDate As Date
    Dim strReminderFullDate As String

```

```
strReminderFullDate = cbxDay.Text & "/" & cbxMonth.Text & "/"  
& cbxYear.Text & " " & cbxHour.Text & ":" & cbxMinute.Text & ":00"
```

```
If cbxReminder.Text = "0 minute" Then  
    strPopOutFullDate = strReminderFullDate  
ElseIf cbxReminder.Text = "5 minutes" Then  
    PopOutFullDate = DateAdd(DateInterval.Minute, -5,  
CDate(strReminderFullDate))  
    strPopOutFullDate = CStr(PopOutFullDate)  
ElseIf cbxReminder.Text = "10 minutes" Then  
    PopOutFullDate = DateAdd(DateInterval.Minute, -10,  
CDate(strReminderFullDate))  
    strPopOutFullDate = CStr(PopOutFullDate)  
ElseIf cbxReminder.Text = "15 minutes" Then  
    PopOutFullDate = DateAdd(DateInterval.Minute, -15,  
CDate(strReminderFullDate))  
    strPopOutFullDate = CStr(PopOutFullDate)  
ElseIf cbxReminder.Text = "30 minutes" Then  
    PopOutFullDate = DateAdd(DateInterval.Minute, -5,  
CDate(strReminderFullDate))  
    strPopOutFullDate = CStr(PopOutFullDate)  
ElseIf cbxReminder.Text = "1 hour" Then  
    PopOutFullDate = DateAdd(DateInterval.Hour, -1,  
CDate(strReminderFullDate))  
    strPopOutFullDate = CStr(PopOutFullDate)  
ElseIf cbxReminder.Text = "2 hours" Then  
    PopOutFullDate = DateAdd(DateInterval.Hour, -2,  
CDate(strReminderFullDate))  
    strPopOutFullDate = CStr(PopOutFullDate)  
ElseIf cbxReminder.Text = "3 hours" Then  
    PopOutFullDate = DateAdd(DateInterval.Hour, -3,  
CDate(strReminderFullDate))  
    strPopOutFullDate = CStr(PopOutFullDate)  
ElseIf cbxReminder.Text = "4 hours" Then  
    PopOutFullDate = DateAdd(DateInterval.Hour, -4,  
CDate(strReminderFullDate))  
    strPopOutFullDate = CStr(PopOutFullDate)  
ElseIf cbxReminder.Text = "5 hours" Then  
    PopOutFullDate = DateAdd(DateInterval.Hour, -5,  
CDate(strReminderFullDate))  
    strPopOutFullDate = CStr(PopOutFullDate)  
ElseIf cbxReminder.Text = "6 hours" Then  
    PopOutFullDate = DateAdd(DateInterval.Hour, -6,  
CDate(strReminderFullDate))  
    strPopOutFullDate = CStr(PopOutFullDate)  
ElseIf cbxReminder.Text = "7 hours" Then  
    PopOutFullDate = DateAdd(DateInterval.Hour, -7,  
CDate(strReminderFullDate))  
    strPopOutFullDate = CStr(PopOutFullDate)  
ElseIf cbxReminder.Text = "8 hours" Then  
    PopOutFullDate = DateAdd(DateInterval.Hour, -8,  
CDate(strReminderFullDate))  
    strPopOutFullDate = CStr(PopOutFullDate)  
ElseIf cbxReminder.Text = "9 hours" Then  
    PopOutFullDate = DateAdd(DateInterval.Hour, -9,  
CDate(strReminderFullDate))  
    strPopOutFullDate = CStr(PopOutFullDate)
```

```

        ElseIf cbxReminder.Text = "10 hours" Then
            PopOutFullDate = DateAdd(DateInterval.Hour, -10,
CDate(strReminderFullDate))
            strPopOutFullDate = CStr(PopOutFullDate)
        ElseIf cbxReminder.Text = "11 hours" Then
            PopOutFullDate = DateAdd(DateInterval.Hour, -11,
CDate(strReminderFullDate))
            strPopOutFullDate = CStr(PopOutFullDate)
        ElseIf cbxReminder.Text = "Half day" Then
            PopOutFullDate = DateAdd(DateInterval.Hour, -12,
CDate(strReminderFullDate))
            strPopOutFullDate = CStr(PopOutFullDate)
        ElseIf cbxReminder.Text = "18 hours" Then
            PopOutFullDate = DateAdd(DateInterval.Hour, -18,
CDate(strReminderFullDate))
            strPopOutFullDate = CStr(PopOutFullDate)
        ElseIf cbxReminder.Text = "1 day" Then
            PopOutFullDate = DateAdd(DateInterval.Day, -1,
CDate(strReminderFullDate))
            strPopOutFullDate = CStr(PopOutFullDate)
        ElseIf cbxReminder.Text = "2 days" Then
            PopOutFullDate = DateAdd(DateInterval.Day, -2,
CDate(strReminderFullDate))
            strPopOutFullDate = CStr(PopOutFullDate)
        ElseIf cbxReminder.Text = "3 days" Then
            PopOutFullDate = DateAdd(DateInterval.Day, -3,
CDate(strReminderFullDate))
            strPopOutFullDate = CStr(PopOutFullDate)
        ElseIf cbxReminder.Text = "4 days" Then
            PopOutFullDate = DateAdd(DateInterval.Day, -4,
CDate(strReminderFullDate))
            strPopOutFullDate = CStr(PopOutFullDate)
        ElseIf cbxReminder.Text = "1 week" Then
            PopOutFullDate = DateAdd(DateInterval.Day, -7,
CDate(strReminderFullDate))
            strPopOutFullDate = CStr(PopOutFullDate)
        ElseIf cbxReminder.Text = "2 weeks" Then
            PopOutFullDate = DateAdd(DateInterval.Day, -14,
CDate(strReminderFullDate))
            strPopOutFullDate = CStr(PopOutFullDate)
        End If
    End Sub

```

```

Public Sub SaveReminder()
    'Refresh the dataset
    DsBMS.Clear()
    totalRows = odaBMS.Fill(DsBMS, "tblReminder")

    'Call TraceLargestID
    TraceLargestID()
    'Increase largestID by 1
    largestID += 1

    'DataRow to create new record entry in tblReminder
    Dim NewRecord As DataRow

    'Create a new entry inside tblReminder table

```

```

NewRecord = DsBMS.Tables("tblReminder").NewRow
NewRecord("ID") = largestID
NewRecord("Subject") = txtSubject.Text
NewRecord("Location") = txtLocation.Text
NewRecord("DateReminder") = cbxDay.Text & "/" & cbxMonth.Text
& "/" & cbxYear.Text & " " & cbxHour.Text & ":" & cbxMinute.Text &
":00"
NewRecord("DateCreate") = Now
NewRecord("RemindTime") = cbxReminder.Text

'Calculate for when the reminder should be pop up
CalculatePopUpTime()

NewRecord("DatePopOut") = strPopOutFullDate
NewRecord("Description") = txtDescription.Text
NewRecord("Status") = "P"
DsBMS.Tables("tblReminder").Rows.Add(NewRecord)

'Update dataset and commit changes
odaBMS.Update(DsBMS)
DsBMS.AcceptChanges()
End Sub

Public Sub EditReminder()
'Refresh the dataset
DsBMS.Clear()
totalRows = odaBMS.Fill(DsBMS, "tblReminder")

'DataRow to edit record entry in tblReminder
Dim FoundRow() As DataRow

FoundRow = DsBMS.Tables("tblReminder").Select("ID='" &
currentID & "'")
If FoundRow.Length = 0 Then
'Ignore, it should be there
Else
'Update the record
FoundRow(0)("Subject") = txtSubject.Text
FoundRow(0)("Location") = txtLocation.Text
FoundRow(0)("DateReminder") = cbxDay.Text & "/" &
cbxMonth.Text & "/" & cbxYear.Text & " " & cbxHour.Text & ":" &
cbxMinute.Text & ":00"
FoundRow(0)("DateCreate") = Now
FoundRow(0)("RemindTime") = cbxReminder.Text

'Calculate for when the reminder should be pop up
CalculatePopUpTime()

FoundRow(0)("DatePopOut") = strPopOutFullDate
FoundRow(0)("Description") = txtDescription.Text
FoundRow(0)("Status") = "P"
End If

'Update dataset and commit changes
odaBMS.Update(DsBMS)
DsBMS.AcceptChanges()
End Sub

```

```

Public Sub DeleteReminder()
    'Refresh the dataset
    DsBMS.Clear()
    totalRows = odaBMS.Fill(DsBMS, "tblReminder")

    'DataRow to delete record entry in tblReminder
    Dim DeleteRow() As DataRow

    DeleteRow = DsBMS.Tables("tblReminder").Select("ID='" &
currentID & "'")
    If DeleteRow.Length = 0 Then
        'Ignore, it should be there
    Else
        'Delete the record
        DeleteRow(0).Delete()
    End If

    'Update dataset and commit changes
    odaBMS.Update(DsBMS)
    DsBMS.AcceptChanges()
End Sub

Public Sub LoadReminder()
    'Refresh the dataset
    DsBMS.Clear()
    totalRows = odaBMS.Fill(DsBMS, "tblReminder")

    If totalRows = 0 Then
        'There is no record exists
        'Filter the UI
        FilterUI()
    Else
        'There is record exist
        'We need to know what is the smallest ID in the database
        TraceSmallestID()
        'We need to know what is the largest ID in the database as
weel
        TraceLargestID()
        'We set the currentID = smallestID
        currentID = smallestID
        'Filter the UI
        FilterUI()
        'Call the LoadData sub
        LoadData(smallestID)
    End If
End Sub

Public Sub LoadData(ByVal Index As Integer)
    'Refresh the dataset
    DsBMS.Clear()
    totalRows = odaBMS.Fill(DsBMS, "tblReminder")

    'Search data for the ID given
    Dim FoundRow() As DataRow
    FoundRow = DsBMS.Tables("tblReminder").Select("ID='" & Index &
""")

```

```

If FoundRow.Length = 0 Then
    'There is no record, do something here
Else
    'Display the data accordingly
    txtSubject.Text = FoundRow(0)("Subject")
    txtLocation.Text = FoundRow(0)("Location")

    'For date and time
    Dim strFullDate As String
    Dim strDate As String
    Dim strTime As String
    Dim strTemp As String
    strFullDate = FoundRow(0)("DateReminder")
    If Not strFullDate Is System.DBNull.Value Then
        strDate = strFullDate.Substring(0,
strFullDate.IndexOf(" "))
        strTime = strFullDate.Substring(strFullDate.IndexOf("
") + 1)

        'For date
        strTemp = strDate.Substring(0, 2)
        cbxDay.SelectedItem = strTemp
        strTemp = strDate.Substring(3, 2)
        cbxMonth.SelectedItem = strTemp
        strTemp = strDate.Substring(strDate.LastIndexOf("/") +
1)

        cbxYear.Text = strTemp

        'For time
        strTemp = strTime.Substring(0, strTime.IndexOf(":"))
        cbxHour.SelectedItem = strTemp
        strTemp = strTime.Substring(strTime.IndexOf(":") + 1,
2)

        cbxMinute.Text = strTemp
    End If

    cbxReminder.SelectedItem = FoundRow(0)("RemindTime")
    txtDescription.Text = FoundRow(0)("Description")
End If
End Sub

```

```

Private Sub frmReminder_Load(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Load
    ReloadFlag = True
    'Set largestID and currentID = 0 first
    largestID = 0
    currentID = 0
    'Set smallestID = 1 first
    smallestID = 1
    ReloadFlag = False

    'First check what type of items need to be displayed
    If modBMS.ReminderType = "New" Then
        'Hide btnFirst, btnPrevious, btnNext, btnLast and
btnDelete
        btnFirst.Visible = False
        btnPrevious.Visible = False

```

```

        btnNext.Visible = False
        btnLast.Visible = False
        btnDelete.Visible = False
        btnSave.Visible = True

        'Reset all the controls
        txtSubject.Text = ""
        txtLocation.Text = ""
        cbxDay.Text = ""
        cbxMonth.Text = ""
        cbxYear.Text = ""
        cbxReminder.Text = ""
        cbxHour.Text = ""
        cbxMinute.Text = ""
        txtDescription.Text = ""
    ElseIf modBMS.ReminderType = "Edit" Then
        'Show all
        btnFirst.Visible = True
        btnPrevious.Visible = True
        btnNext.Visible = True
        btnLast.Visible = True
        btnDelete.Visible = True

        'Need to display data in the database
        LoadReminder()
    Else
        'Show all
        btnFirst.Visible = True
        btnPrevious.Visible = True
        btnNext.Visible = True
        btnLast.Visible = True
        btnDelete.Visible = True
        btnSave.Visible = True

        'It is EditSpecial type
        'We need to know what is the smallest ID in the database
        TraceSmallestID()
        'We need to know what is the largest ID in the database as
weel
        TraceLargestID()
        'Extract the ID
        currentID =
CInt(modBMS.ReminderType.Substring(modBMS.ReminderType.LastIndexOf(".")
) + 1))
        FilterUI()
        LoadData(currentID)
    End If
End Sub

Private Sub frmReminder_Closing(ByVal sender As Object, ByVal e As
System.ComponentModel.CancelEventArgs) Handles MyBase.Closing
    e.Cancel = True
    ReloadFlag = False
    Me.Hide()
End Sub

```



```

Private Sub frmReminder_Activated(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles MyBase.Activated
    If ReloadFlag = False Then
        Me.ReloadForm()
        ReloadFlag = True
    End If
End Sub

Private Sub btnSave_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnSave.Click
    'Check for validity of data
    If txtSubject.Text = "" Then
        MsgBox("Please fill in Reminder Subject.",
MsgBoxStyle.Information Or MsgBoxStyle.OKOnly, "Incomplete
Information")
        txtSubject.Focus()
        Exit Sub
    End If
    If cbxDay.Text = "" Or cbxMonth.Text = "" Or cbxYear.Text = ""
Then
        MsgBox("Please fill in Reminder Date.",
MsgBoxStyle.Information Or MsgBoxStyle.OKOnly, "Incomplete
Information")
        cbxDay.Focus()
        Exit Sub
    End If
    If cbxHour.Text = "" Or cbxMinute.Text = "" Then
        MsgBox("Please fill in Reminder Time.",
MsgBoxStyle.Information Or MsgBoxStyle.OKOnly, "Incomplete
Information")
        cbxHour.Focus()
        Exit Sub
    End If
    If cbxReminder.Text = "" Then
        MsgBox("Please fill in when do you want to be reminded.",
MsgBoxStyle.Information Or MsgBoxStyle.OKOnly, "Incomplete
Information")
        cbxReminder.Focus()
    End If

    'Check what type of reminder
    If modBMS.ReminderType = "New" Then
        'If it is new reminder, call SaveReminder sub routine
        SaveReminder()
        'RaiseEvent that reminder has been modified
        RaiseEvent ReminderUpdated()
        'Hide the form
        Me.Hide()
    Else
        'If it is edit or edit special reminder, call EditReminder
sub routine
        'Confirm with user first
        Dim Ans As Integer
        Ans = MsgBox("Do you really want to update the data?",
MsgBoxStyle.Question Or MsgBoxStyle.YesNo, "Edit Confirmation")
        If Ans = vbYes Then
            'If yes, then edit the data

```

```

        EditReminder()
        'RaiseEvent that reminder has been modified
        RaiseEvent ReminderUpdated()
    Else
        'If no, just exit the event
        Exit Sub
    End If
End If
End Sub

Private Sub btnDelete_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnDelete.Click
    'Confirm with user first
    Dim Ans As Integer
    Ans = MsgBox("Do you really want to delete this reminder?",
MsgBoxStyle.Question Or MsgBoxStyle.YesNo, "Delete Confirmation")
    If Ans = vbYes Then
        'If yes, then delete the data
        DeleteReminder()
        'Load back the reminder
        LoadReminder()
        'RaiseEvent that reminder has been modified
        RaiseEvent ReminderUpdated()
    Else
        'If no, just exit the event
        Exit Sub
    End If
End Sub

Private Sub btnFirst_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnFirst.Click
    'Set currentID = smallestID
    TraceSmallestID()
    currentID = smallestID
    'Filter the UI
    FilterUI()
    'Call the LoadData sub
    LoadData(currentID)

    'MsgBox("Current ID: " & currentID & vbCrLf & "Smallest ID : "
& smallestID & vbCrLf & "Largest ID : " & largestID)
End Sub

Private Sub btnPrevious_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles btnPrevious.Click
    'Decrease currentID by 1
    currentID -= 1
    'Check the validity of the ID being appointed
    CheckIDValidity("Decrement")
    'Filter the UI
    FilterUI()
    'Call the LoadData sub
    LoadData(currentID)

    'MsgBox("Current ID: " & currentID & vbCrLf & "Smallest ID : "
& smallestID & vbCrLf & "Largest ID : " & largestID)
End Sub

```

```

Private Sub btnNext_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnNext.Click
    'Increase currentID by 1
    currentID += 1
    'Check the validity of the ID being appointed
    CheckIDValidity("Increment")
    'Filter the UI
    FilterUI()
    'Call the LoadData sub
    LoadData(currentID)

    'MsgBox("Current ID: " & currentID & vbCrLf & "Smallest ID : "
& smallestID & vbCrLf & "Largest ID : " & largestID)
End Sub

Private Sub btnLast_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnLast.Click
    'Set currentID = largestID
    TraceLargestID()
    currentID = largestID
    'Filter the UI
    FilterUI()
    'Call the LoadData sub
    LoadData(currentID)

    'MsgBox("Current ID: " & currentID & vbCrLf & "Smallest ID : "
& smallestID & vbCrLf & "Largest ID : " & largestID)
End Sub

Private Sub cbxMinute_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles cbxMinute.TextChanged
    'Check for the validity of the minute
    If Val(cbxMinute.Text) > 60 Then
        MsgBox("Enter the range value up to 60 only",
MsgBoxStyle.Information Or MsgBoxStyle.OKOnly, "Invalid data")
        cbxMinute.Text = ""
        cbxMinute.Focus()
    End If
End Sub

End Class

```