Control Theory and Informatics
ISSN 2224-5774 (Paper) ISSN 2225-0492 (Online)
Vol.3, No.5, 2013

Design of A Mobile Phone Data Backup System

AREGBESOLA Moses Kehinde[1] and Osunade A.O.[2]
College of Information and Communication Technology, Salem University, Lokoja, Kogi State, Nigeria[1]
Department of Computer Science, University of Ibadan, Ibadan, Oyo State, Nigeria[2]
kennie747@yahoo.com[1]

Abstract
This study examined how mobile phones work, the design of the graphical user interface, the design of the application and the design of the database used for the K-Backup system. This application backs up mobile phone and Subscribers Identification Module (SIM) data (contacts and SMS) on the computer storage. It also allows for easy retrieval and restoration of the data to phone memory when the need arises. The objectives considered in the design included: user-friendly graphical user interface (GUI) for user interaction; modular subroutines, methods and functions that retrieves data from the phone and stores it on the computer memory; modular subroutines, methods and functions that retrieves data from the SIM and stores it on the computer memory; modular subroutines and functions that send data from the computer back to the phone; portable database for storing and retrieving the data.
Keywords: Mobile Phone, Data Backup, System Design, GSM, Phone Data.

1. Introduction

Mobile phones come in different shapes and sizes. Owners of GSM phones are sometimes classed by the type of phone(s) they own and the functionalities the phones can perform. The demand for functionalities that come with GSM phones has hence increased significantly. The major striking features that most users clamor for apart from the phone having digital camera/recording facilities is the ability of their phones to store more data – or at least have some sort of extended memory - , display a high level of security and provide for backup of phone data (in case of any mishap to the phone). As more and more functionalities are added to these phones, the more expensive they become. Hence these days, we find phones costing hundreds of thousands of naira. Not all can afford phones this expensive. Everyone wants to maximize and improve on what he already has. The average man is in search of a means by which he can extend the functionalities that port naturally with his phone.
There is the risk of loosing all of ones contacts and valuable text messages in situations of Phone/SIM loss, damage, or erasure by mistake due to careless use. It is hence very necessary to back up the data in the phonebook of a mobile phone/SIM.
One can manually write out all these data from his phone. This would however be very time consuming and tedious. Besides, it would be extremely difficult to keep this manual record up to date since the user will need to browse through the entire contents of the phone each time he plans to make an update of his record.
It is for this reason that this data backup tool was designed and developed to automate the backup of mobile phone data. Details of this software are discussed in subsequent chapters of this work.

2. Literature Review

In spite of its current popularity, mobile phones have existed long before GSM was conceived. In telecommunication, a wireless phone**,** cellular mobile, cell phone or mobile phone, is a mobile communications system that uses a combination of radio wave transmission and conventional telephone switching to permit telephone communication to and from mobile users within a specified area.
The term does not comprise the so-called portable phone or cordless phone, which is associated with a fixed telephone landline and can only be operated close to (less than 100 meters of) its base station, such as in and around the house. The term cell phone applies specifically to mobile phones which use a cellular network; satellite phones are also mobile phones, but not cellular [2, 9].

In cellular mobile systems, large geographical areas are segmented into many smaller areas, i.e., cells, each of which has its own radio transmitters and receivers and a single controller interconnected with the public switched telephone network. Each cell site has a range of 3-5 miles and overlaps other cell sites. All of the cell sites are connected to one or more cellular switching exchanges which can detect the strength of the signal received from the telephone. As the telephone user moves or roams from one cell area to another, the exchange automatically switches the call to the cell site with the strongest signal [2, 9].

All cell phones have special codes associated with them. These codes are used to identify the phone, the phone's owner and the service provider [8]. They are:

- Electronic Serial Number (ESN)
- Mobile Identification Number (MIN)
- System Identification Code (SID)

While the ESN is considered a permanent part of the phone, both the MIN and SID codes are programmed into the phone when you purchase a service plan and have the phone activated [2, 9].

3. Requirement Analysis and Design

A series of tools and methodologies were used in the design of the system. The overall system design, organizational flow and database structure are clearly represented.

3.1 Analysis of Existing Systems

Most people that currently have a backup of there mobile phone data acquired these data manually from the phone. By manually, it is meant that they scroll through the mobile phone and with the aid of a pen and several pieces of paper copy out the data displayed on the Liquid Crystal Display (LCD). Many of the other existing systems are mere modifications of this manual technique.

This technique obviously promises to be highly time consuming and stressful. Thus, most people would rather go without a backup for the mobile phone data than go through the stress of backing up the data manually.

3.2 Requirement Analysis of the New System

The system after it has been developed must satisfy these requirements for it to be considered satisfactory. The analysis of the requirements expected of the system to be developed is as follows:

The system to be developed must first and foremost be able to fully identify any mobile phone attatched to it. The system should then be able to read the mobile phones phonebook which consist of contact names and their phone numbers, and transfer them into the computer memory. It should also be able to read text messages received through the short message service (SMS), and transfer them into the computer memory. The data read from the phone must be displayed for the user to see, and saved only if the user chooses to do so. The data stored on the computer memory should be retrievable and readable in the form of a phonebook and direct access list for both the contacts and text messages respectively. The user of the system should be able to transfer these data stored on the computer unto a mobile phone when he wants to.

Other functionalities expected of the system is that it should be able to allow the user to send text messages from the system to any network of his choice, using SMS offered by the subscriber's network. The system should support multiple users.

3.3 Design of the New System

This section will be considering the design of the system using different design models to project the system from different angles of view. The different models used will allow the entire system to be seen when they are read with respect to the perspective represent. Naturally, different models are usually best suited to different views of a system under development.

3.3.1 Context Diagram

It is used to represent pictorially, the scope and boundaries of the area under study. The purpose is to identify what is to be included in the study. The area to be studied is shown as a single circle in the center of the diagram. The system is not isolated but interacts with other entities which are shown as rectangles on the context diagram.

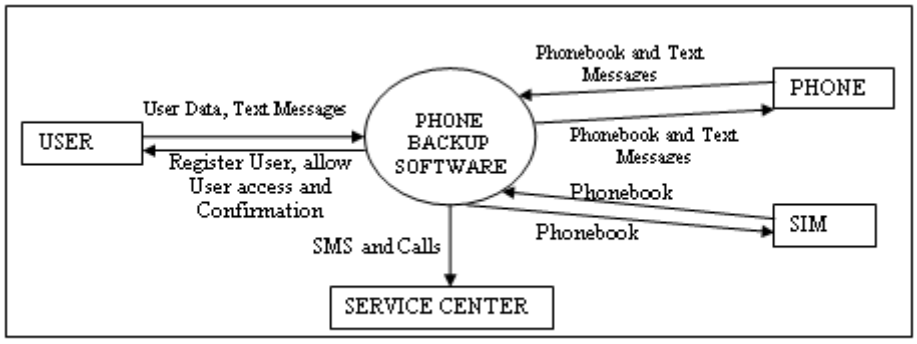

Fig. 1: The Context Diagram for the system

3.3.2 Data Flow Diagram (DFD)

It is a graphic representation of the system that shows data flows to, from and within the system, processing functions that change the data in some manner, and the storage of the data.
The data flow diagram can be used to represent systems at different levels of depth. The level 0 (zero) data flow diagram is the topmost level at which a system can be represented and has the least complexity.

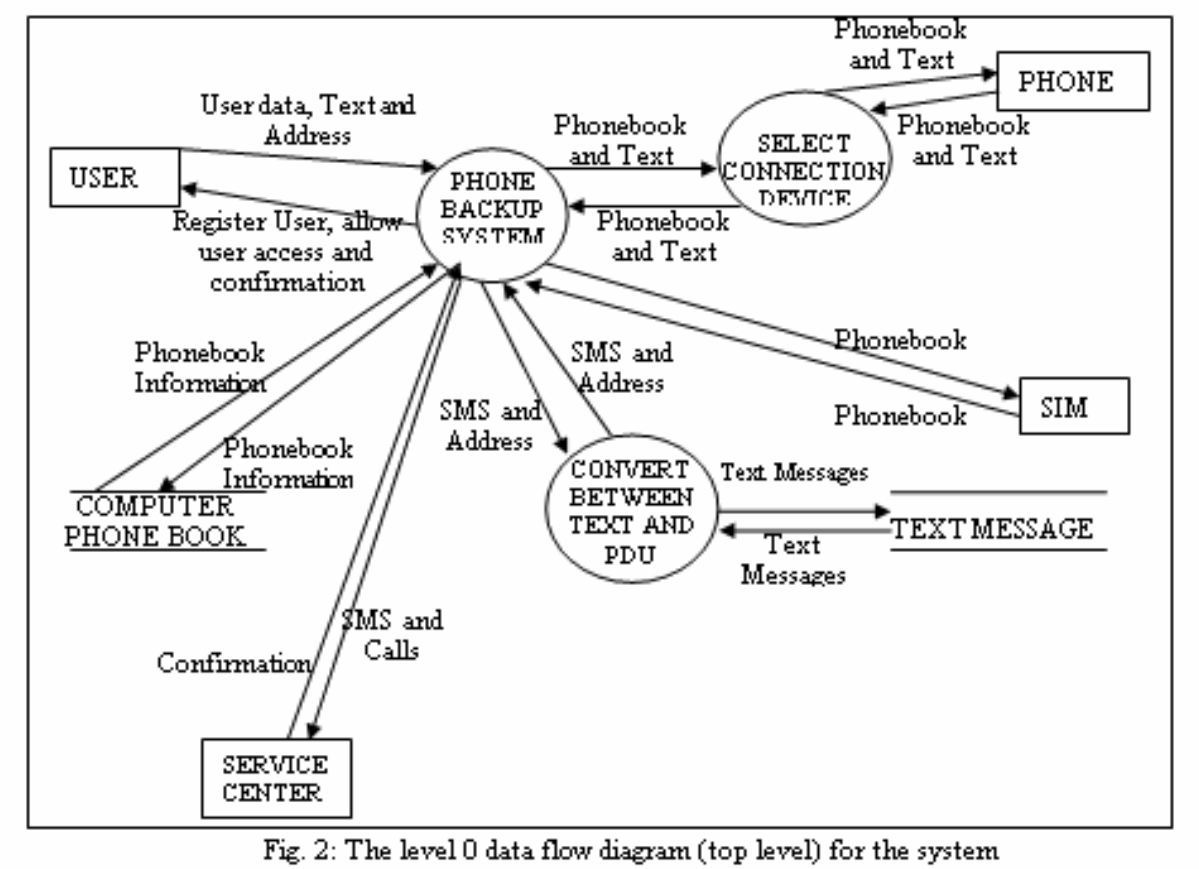

Fig. 2: The level 0 data flow diagram (top level) for the system

3.3.3 The Unified Modeling Language (UML) Design of the system

The Unified Modeling Language is a notational approach that is popular for describing object-oriented (OO) solutions. It can be tailored to fit different development situations and software life cycles [1, 5].
UML can be used to visualize, specify, or document a problem. UML diagrams include the dynamic view of the system, lists of activities, interactions, sequences, collaboration, and state diagrams.
The UML diagrams also show the static view of the system depicted by class, package and deployment diagrams.

3.3.3.1 The State Diagram

The state diagram shows all possible states that an object can take. The change from one state to another is triggered by a message (representing an event) that is sent from one object to another. It is usually necessary when a class has many state changes [1, 5].

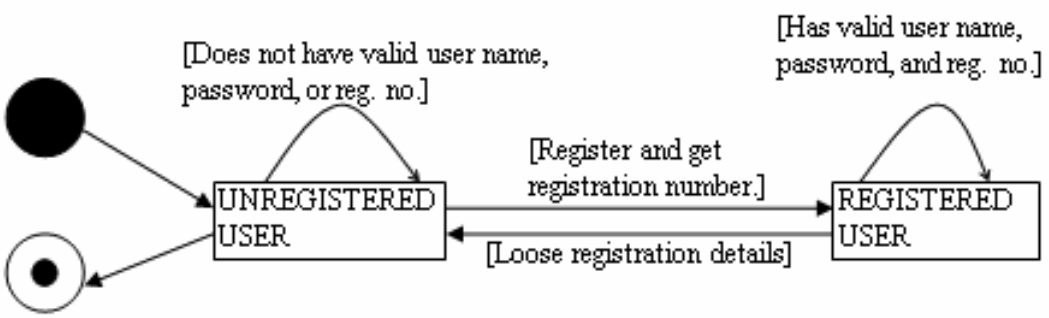

Fig. 3: State Diagram for User Class

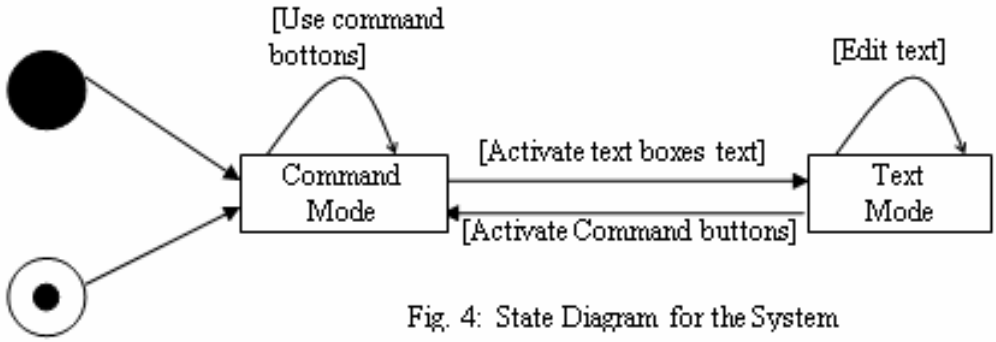

Fig. 4: State Diagram for the System

3.3.3.2 Class Diagram of the New System

The UML class diagrams are used to document the static structure of the system. That is, it shows the classes [4] being used and how they are related, but not how they interact to achieve particular behaviors. A class diagram also shows other aspects of static structure such as packages [1].

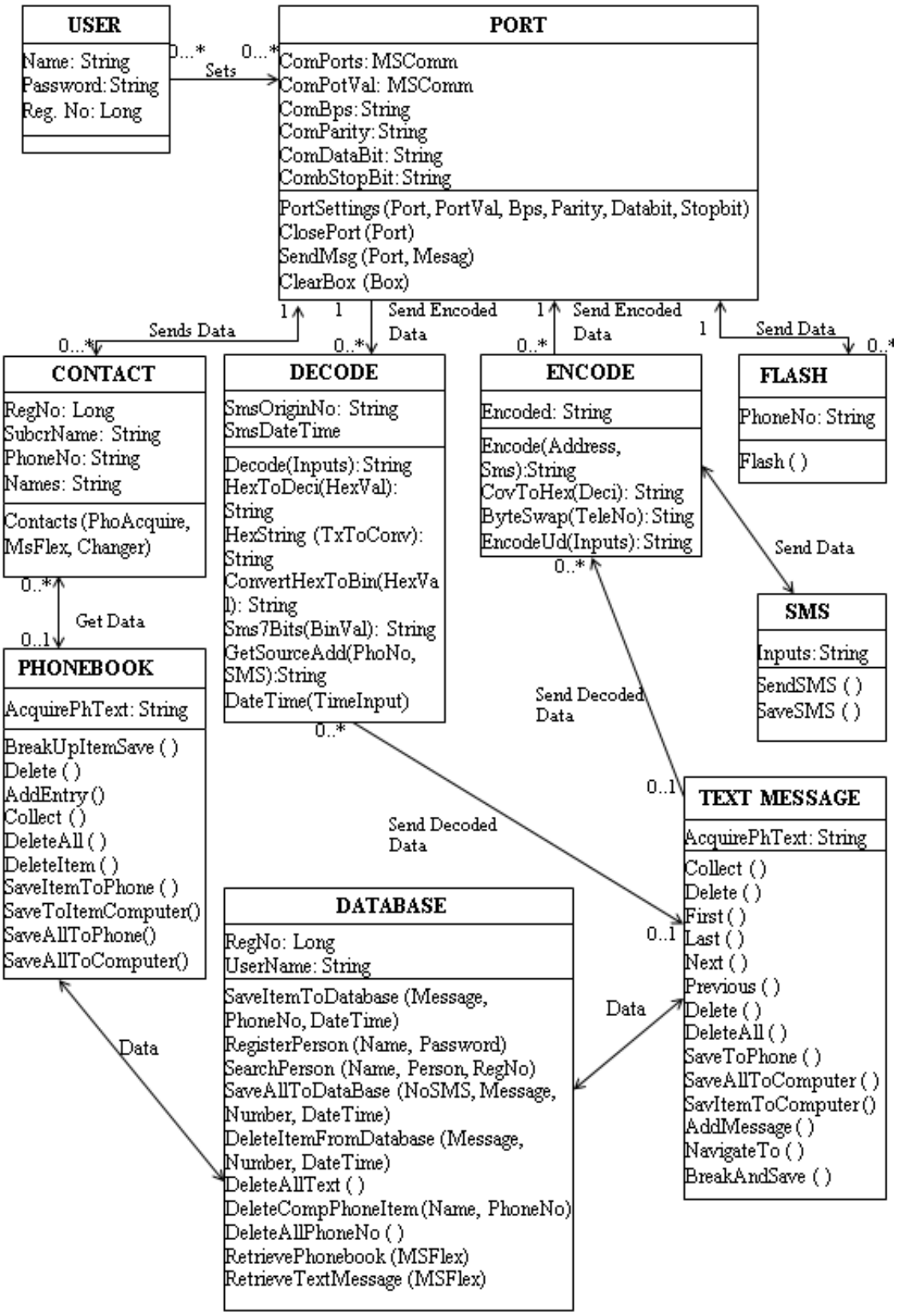

Fig. 5: Class Diagram

3.3.3.3 The Use Case Diagram

The sequence diagram describes the set of sequence of actions including variants, that a system performs that yields an observable result of value to a particular actor. A use case describes what a system (Subsystem, class, or interface) does, but it does not specify how it does it. This use case represents a functional requirement of the system as a whole [5].

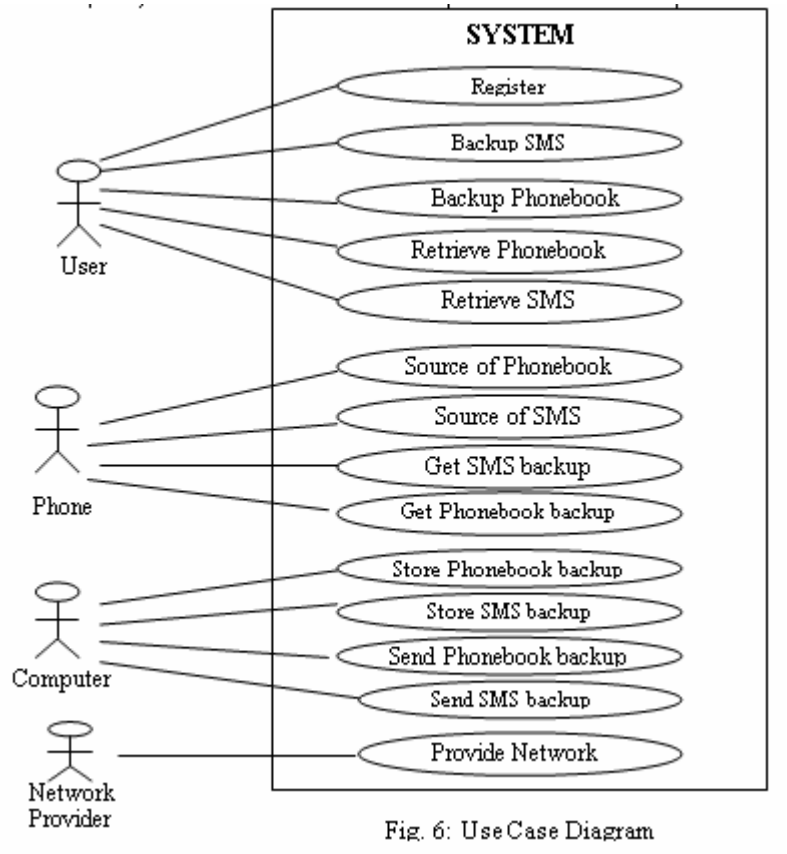

Fig. 6: Use Case Diagram

The interactions among classes are illustrated using interaction diagrams. The interaction diagrams are of two types; Sequence and Collaboration Diagrams.

3.3.3.4 The Package Diagram

Package diagram gives a high-level overview of the system and notes the high-level dependencies [11]. The package diagram below shows how the classes are logically divided into modules. The dashed arrows show the package dependencies.

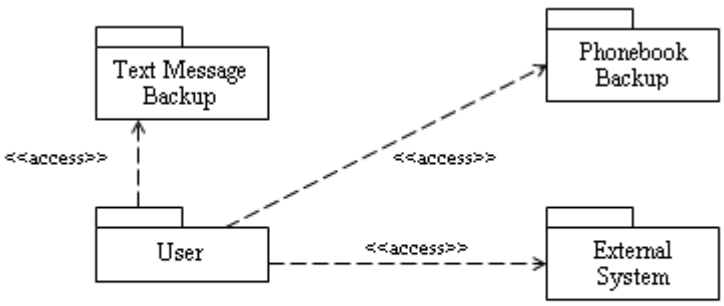

Fig 7: Package Diagram for the System

3.3.3.5 The Sequence Diagram

The Sequence diagram shows how messages flow from one object to another. It shows the sequence in which activities or behaviors occur.

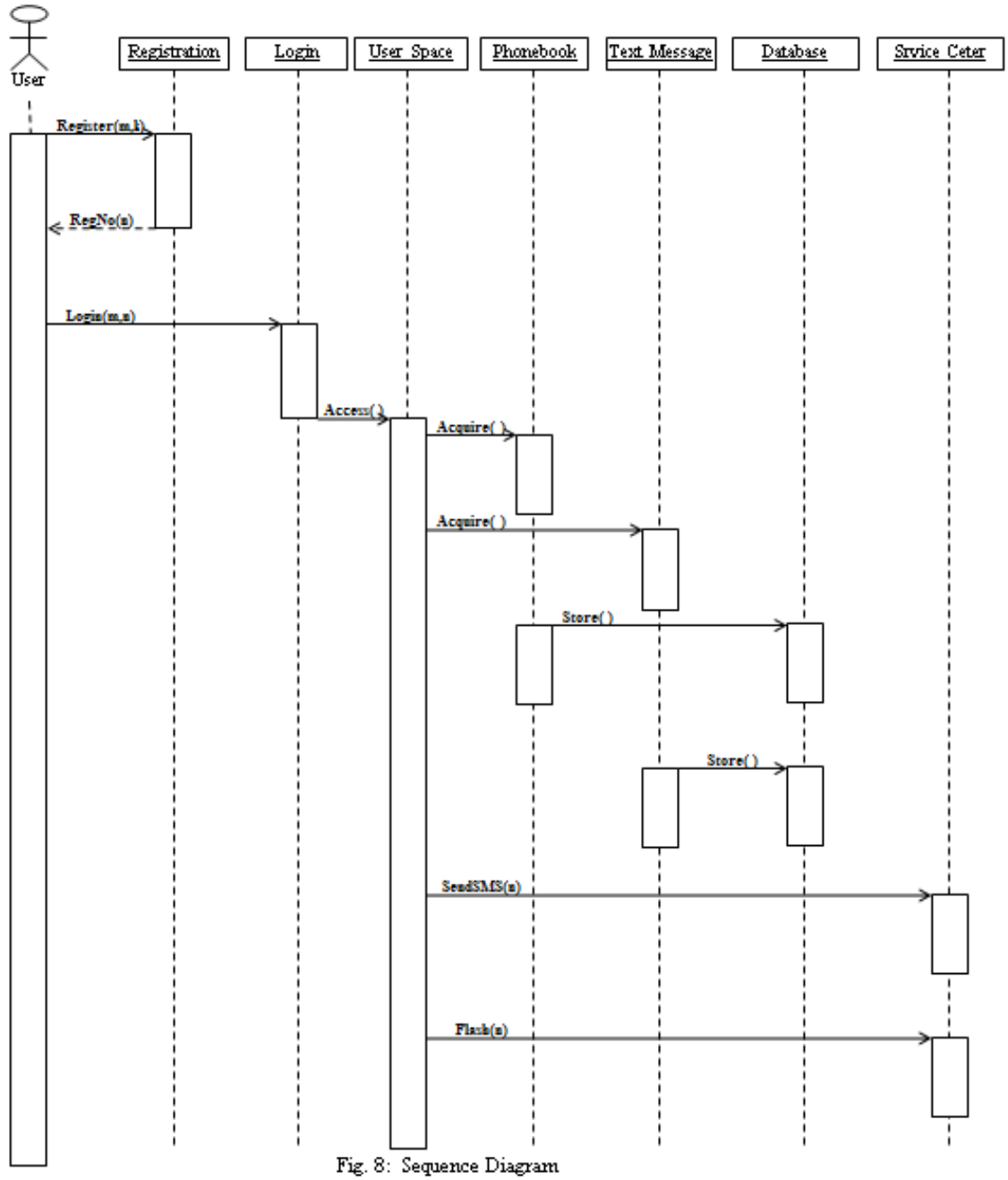

Fig. 8: Sequence Diagram

3.3.3.6 The Collaboration Diagram

The collaboration Diagram uses objects and sequence information to show how the objects are connected statically.

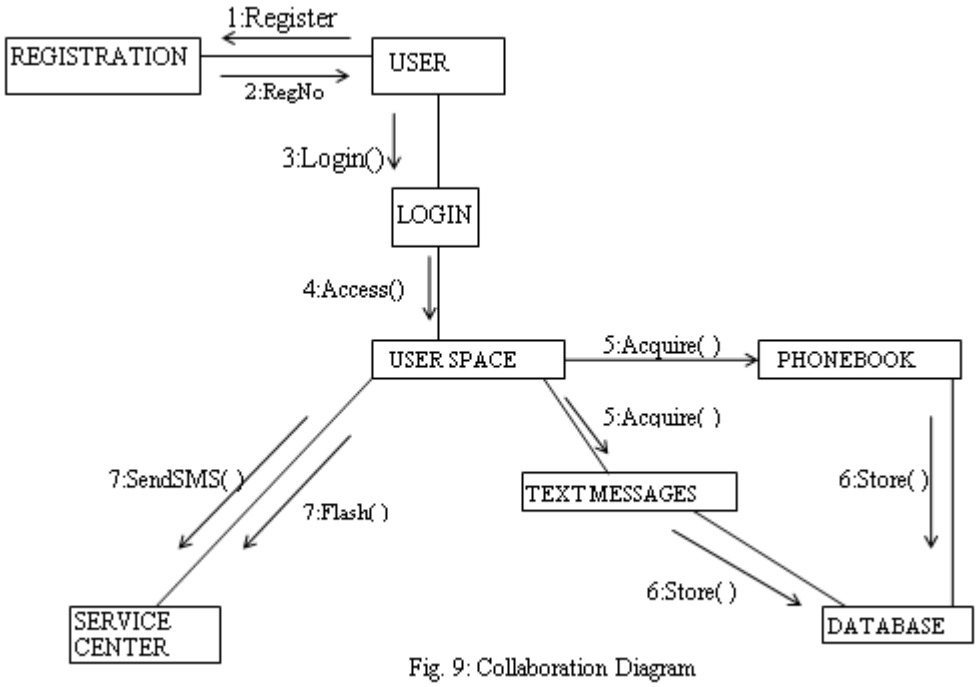

Fig. 9: Collaboration Diagram

3.3.3.7 The Activity Diagram

The activity diagrams below display all activities that can occur in the system as the values of an object change [5].

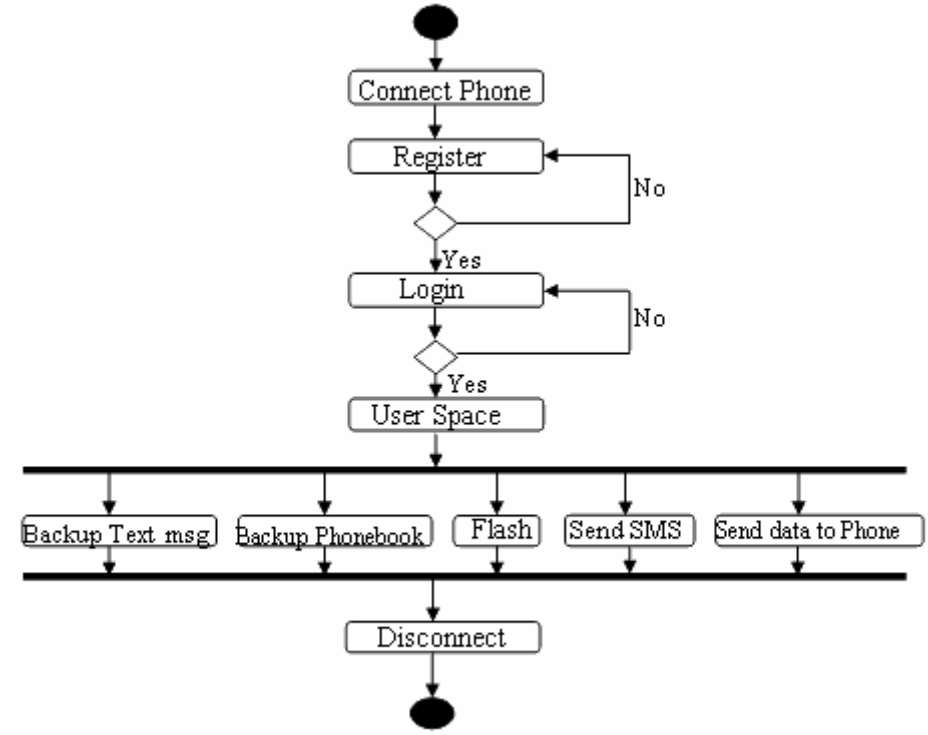

Fig 10: Activity Diagram for the System

3.3.3.8 The Component Model

The component model shows the dependencies between parts of the code. It is primarily of interest to designers and maintainers of the system, and forms part of the development view.

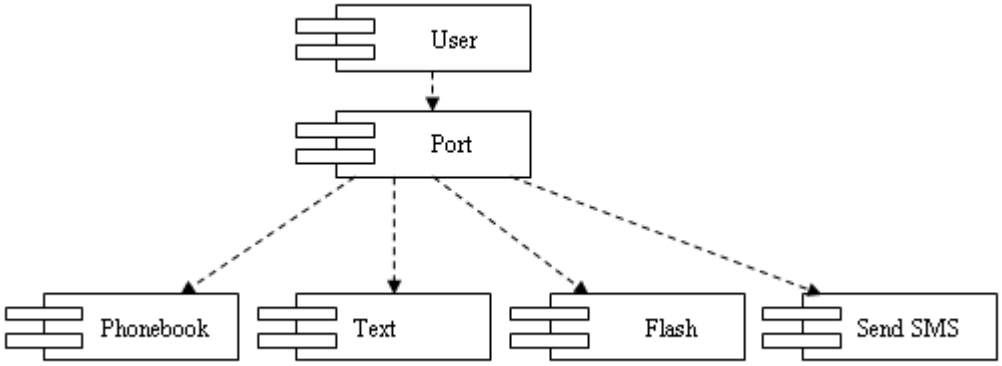

Fig. 11: Component

3.3.3.9 The Deployment Diagram

The deployment model shows the structure of the runtime system: which parts run on which processors and how the hardware is configured to provide necessary resources. It contributes to both the physical view and the process view [10]. The deployment diagram shows:

- The physical communication links between hardware items (machines and other resources)
- The relationship between physical machine and processes that run on them

The physical system consists of nodes with association between them. A node may be a processor or some other devices which provide services such as the mobile phone in this case.

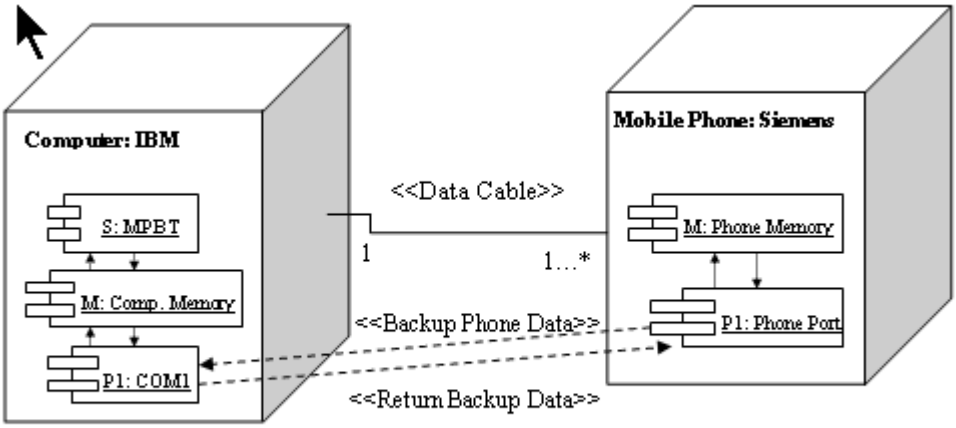

Fig 12: Deployment Diagram

3.3.4 Database Design of the System

3.3.4.1 Entity Relation Model

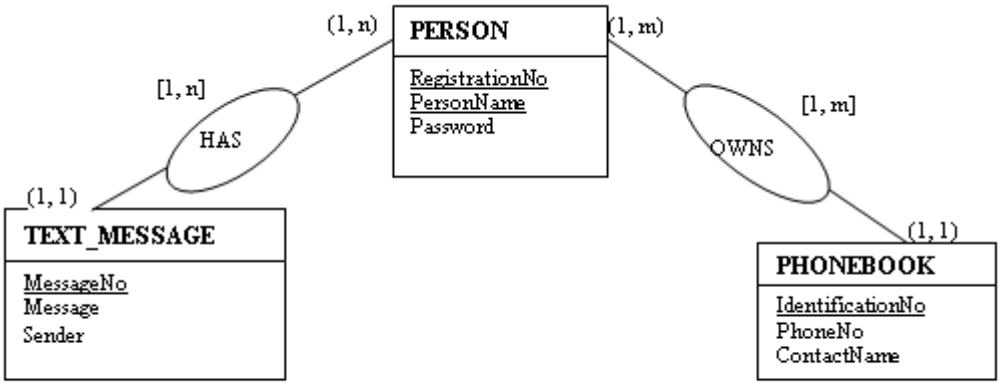

Fig 13: Entity Relation Model

3.3.4.2 Relational Model

Table 1: Relational Model - Person

RegistrationNo	PersonName	Password

Table 2: Relational Model – Phonebook

IdentificationNo	PhoneNo	ContactName

Table 3: Relational Model – Text Message

MessageNo	Message	Sender

3.3.4.3 Relational Translation

Person (RegistrationNo, PersonName, Password)

Phonebook (IdentificationNo, PhoneNo, ContactName, RegistrationNo, PersonName)

Text_Message (MessageNo, Message, Sender, RegistrationNo, PersonName,)

3.3.4.4 Graph of Relation

The Graph of Relation shows the relationship between different entities. Arrow-head lines are used to show how attributes of one entity are linked with attributes of some other entity [3].

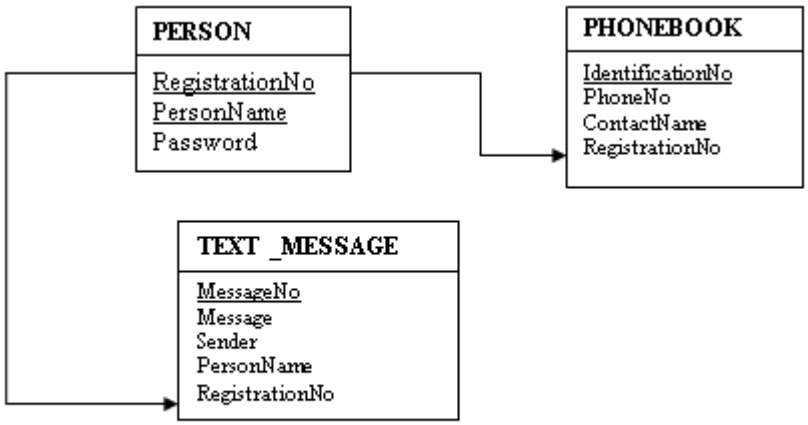

Fig 15: Graph of Relation

2.3.4.5 Dictionary of Attributes

The Dictionary of attribute is a form of documentation that shows all attribute, their data types, the entity to which they belong and a description [3].

Table 4: The dictionary of Attributes for the system

S/N	Attribute Name	Data Type (Domain)	Entity	Description
1	RegistrationNo	Number	Person	Identifies each subscriber
2	PersonName	Text(20)	Person	Subscriber name
3	Password	Text(20)	Person	Subscriber password
4	MessageNo	Number	Text_Message	Identifies each text message
5	Message	Text(160)	Text_Message	The message
6	Sender	Number	Text_Message	The number that sent the message
7	IdentificationNo	Number	Phonebook	Identifies each phone book entry
8	PhoneNo	Number	Phonebook	The contact phone numbers
9	ContactName	Text(20)	Phonebook	The contact names

3.4 Input of the System

The system input would be from the memory of the SIM or mobile phone connected to the computer, or straight from the keyboard through the graphical user interface of the system resident on the computer.
The mobile phone can be connected to the system at any stage of operation. Once the phone is connected, the commands issued out by pressing buttons on the graphical user interface are immediately executed. If no phone is connected, commands related to the mobile phone will be ignored.

3.5 Output of the System

The output of the system is the phonebook backup and text message backup that are retrieved from the mobile phone, and stored on the computer memory. Copies of these backup can latter be transferred to the mobile phone when the need arises. Also as output is the text messages that are and sent from the system through the short message service offered by the network operator that supplied the SIM. Finally is the “flashing” (make and break calls) capability which is also through the network operator that supplied the SIM.

4. Summary, Conclusion and Recommendation

4.1 Summary of the Project

This application has been carefully developed to standard. It is highly scalable and robust. By this, mobile phone users can easily and very efficiently store and retrieve the contacts and text messages on mobile phones. At the convenience of there system, users can easily type text messages using the keyboard which is much faster and

convenient. They can also decide to send SMS and/or simply save the message to the phone. With this system as well, 'Flashing' has been made more accurate.

This work has been strategically laid out and organized to facilitate understanding of basic concepts involved. Chapter One introduced the system as a whole. Following the work all the way to Chapter Four (which discusses the System Implementation) must have equipped any user with some fundamental understanding of the system which includes: what mobile phones are; SIM; the need for backup; technologies involved in data transfer between mobile phone and computer; etc.

Taking a chapter at a time, the summaries derived are as follows:

Chapter one showed the justification, aims and objectives for the development of this system.

Chapter two was a review of existing literatures that are related to this work.

In chapter three, the requirement analysis and design were taken care of. This chapter saw different design models such as the context diagram, dataflow diagram, and the UML design which consisted of several different parts. Data modeling concluded it all.

In chapter four, system implementation and findings were handled. Contained here include system testing, system installation, system documentation and system maintenance.

4.2 Conclusion

This system has undergone a series of unit testing, integration testing, and installation testing. It has been found to perform up to expectation on all grounds. All functions, subroutines, modules, classes and objects are fully functional and produce the expected results - behaving as expected. This system actually works!

The system has been designed in such a way as to allow for rapid expansion or upgrade. By this I mean to say that, once new classes are built (that handle things like to connect via Infrared port etc), allowances have been created within the application were they can be easily plugged and made functional, adding to the current functionalities of the system.

4.3 Recommendation

At this juncture, I observed that this system presently works only for connections made with mobile phones via the serial port.

I hereby recommend that classes are developed or provided, that would allow the system to handle connections established with mobile phones via other data transmission technologies, including connections via the USB port, Bluetooth, Infrared etc.

I also recommend that this system be made web based. By this, you can always reach your backup and other functionalities anytime, any day, and anywhere. Just as long as there is internet access.

REFERENCES

1. Perdita Stevens, Rob Pooley, Using UML Software Engineering with Objects and Components. Editor Addisson-Wesley. June 1999.
2. Cheeseman, David. The pan-European cellular mobile radio system. In R.C.V. Macario, Personal and Mobile Radio Systems. Peter Peregrinus, London, 1991.
3. Prof. Amos David, Data Modeling and Relational Model. Summer School at the department of Computer Science, University of Ibadan. 2004
4. Michael McKelvy, MCSD: Visual Basic 6 Desktop Application Study Guide. Gira Sound 1996.
5. Bernd Oestereich, Developing Software with UML, Addison Wesley, 1999 .
6. Simon Bennett, Steve McRobb and Ray Farmer, Object Oriented systems analysis and design using UML, McGraw-Hill, 1999.
7. Sheri Lawrence Pfleeger, Software Engineering Theory and Practice (Second edition), 2001.
8. Maduka Ifeanyi, Development of a Phonebook Software, Thesis submitted to the Department of Computer Science, University of Ibadan, 2005.
9. Feldmann, M. and Rissen J.P. GSM Network Systems and Overall System Integration. Electrical Communication, 2nd Quarter 1993.
10. Martin Fowler, Kendall Scott, UML Distilled, Addison Wesley, 2000.
11. Grady Broch, James Rambaugh, Ivar Jacobs. The Unified Modeling Language (UML): User Guide (1991) Addison-Wesley. Boston USA.