Mobile Game: A Multiplayer Mobile Game via Bluetooth Technology for Nokia Series 60

By

Ng Yin Ping

Dissertation submitted in partial fulfillment of
the requirements for the
Bachelor of Technology (Hons)
(Information Technology)

JUNE 2004

Universiti Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

Mobile Game: A Multiplayer Mobile Game via Bluetooth Technology for Nokia Series 60

By

Ng Yin Ping

A project dissertation submitted to the
Information Technology Programme
Universiti Teknologi PETRONAS
in partial fulfillment of the requirements for the
BACHELOR OF TECHNOLOGY (Hons)
(INFORMATION TECHNOLOGY)

Approved by,

(Mr. Khairul Shafee Kalid)

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK
June 2004

CERTIFICATION OF ORIGINALITY

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

(NG YIN PING)

ABSTRACT

The Final Year Project developed is *'Mobile Game: A Multiplayer Mobile Game via Bluetooth Technology for Nokia Series 60'*. The existing mobile games are mostly a stand-alone application, which means the game can only be played by a player. Even with the availability of multiplayer mobile games in the market, the maximum number of players of those games is two. The objective of this project is to study and develop a multiplayer (specifically three players) mobile game with Bluetooth technology as the connection means. The first part of this project would be the research and study pertaining to the development of multiplayer mobile game for three players using Bluetooth technology. The second part of the research pertains to the downside of Bluetooth technology and recommendations to improve the situation. The methodology used is the combination of waterfall and prototyping model. A mobile game application is produced as the end-product of this project. Meanwhile, the findings discover the details on the end-product development and also some discussion design issues and challenges as well as limitation of Bluetooth technology.

ACKNOWLEDGEMENT

I would like to take this opportunity to extend my greatest appreciation and thank you to my supervisor of this Final Year Project (FYP), Mr. Khairul Shafee Kalid for his endless support, guidance, encouragement and assistance throughout the entire one semester. Thank you for being such a motivating and supportive mentor that lead me all the way till the completion of this project.

Next, I would like to express my appreciation to the FYP Committee and lecturers from IT/IS department for their guidance and help whenever I needed it.

Adding on to the list is Mr. Foong Wai Seng, the external examiner for this project. Thank you for all your constructive comments and suggestions.

Last but not least, I would like to thank all my friends and family for all the help, guidance and support during the whole period of this FYP. Without your motivation, I would not complete this project successfully. Thanks a lot.

TABLE OF CONTENTS

CHAPTER 3 – METHODOLOGY / PROJECT WORK

CHAPTER 4 - RESULTS AND DISCUSSION

LIST OF FIGURES

LIST OF TABLES

ABBREAVIATIONS AND NOMENCLATURES

PC	Personal Computer
PDA	Personal Digital Assistant
LC	Link Controller
LMP	Link Manager Protocol
HCI	Host Controller Interface
L2CAP	Logical Link Control And Adaptation Protocol
SDP	Service Discovery Protocol
GIAC	General Inquiry Access Code
DIAC	Dedicated Inquiry Access Codes
UI	User Interface

CHAPTER 1
INTRODUCTION

1. 1 BACKGROUND OF STUDY

The evolution of coloured and Java-enabled phones has changed the trend of classic black and white mobile games to the more appealing and more powerful coloured mobile game using Java technology. Adding on to that is the creation of smartphones which implement Symbian OS as their operating system. Smartphone is a powerful device that extends the superb voice functionality of a mobile phone into the realm of data communications.

Mobile games can be either a stand-alone application or a networked application whereby the game can support multi player. Before the existence of mobile phones with Bluetooth technology, a networked mobile game is played by connecting two mobile phones using infrared, which is embedded in certain phone models. However, there are two disadvantages of infrared connection; the number of players for networked game is limited to only two and the phone has to be in line-of-sight mode, which means the infrared port of the two phones has to be pointed to each other all the time during the entire period of the networked game session.

Thus, the current generation of mobile phones comes with Bluetooth technology for an improvised short distance wireless connection, no more than 30 feet. Bluetooth is a specification for using low-power radio to link phones and computers over short distances without wires. Besides that, Bluetooth technology enables mobile phone users to play networked games up to eight players at a time without the need to be in line-of-sight mode.

1.2 PROBLEM STATEMENT

1.2.1 Problem identification

The existing mobile games are mostly a stand-alone application, which means the game can only be played by a player. Even with the availability of networked mobile games (also known as multiplayer mobile games) in the market, the maximum number of players of those games is limited to two.

Besides that, the capability of Bluetooth technology that can support connection of up to eight devices at a time is not fully deployed in mobile game development. Implementing Bluetooth technology in multiplayer mobile games would greatly improve the current capability and excitement of multiplayer mobile games.

1.2.2 Significance of the project

This project will explore the development of a multiplayer mobile game that uses Bluetooth technology as the connection platform for mobile phones. Upon the completion of the project, it can provide an insight of the step-by-step development of a multiplayer mobile game, specifically three players.

1.3 OBJECTIVES AND SCOPE OF STUDY

1.3.1 Objectives

- To identify the step-by-step development of multiplayer mobile game.
- To solve the major problems of multiplayer mobile games via infrared by using Bluetooth technology as the connection means.
- To produce a prototype of a 3-player game application that uses Bluetooth technology.

1.3.2 Scope of Study

The scope of study of this project will be basically divided into three parts. The first portion of this project will focus on research pertaining to the development of multiplayer mobile game for three players via Bluetooth technology specifically for Nokia mobile phone under the Series 60 category. The issue involved in this part of study includes mobile game, which can be divided into single and multiplayer game. Currently, common mobile games categories include Arcade, Puzzle, Action, Sport, and Traditional. The focus would be on multiplayer mobile game and the specific type of game that will be concentrated in this project is traditional game (board game) due to time constraint that it is insufficient to produce other type of more challenging game. Besides that, key design issues and challenges in developing a mobile game and the connectivity of mobile phones during the mobile game session using Bluetooth technology will also be studied.

The second division of the study is the research part in this project. It will look into the limitation of Bluetooth technology that could possibly affect the performance of the devices during the gaming session

The final scope would be producing the end-product. The end-product of this project would be a multiplayer mobile game application that could support three players that uses Bluetooth wireless technology as the connection mode.

CHAPTER 2
LITERATURE REVIEW AND THEORY

2.1 DIFFERENCES BETWEEN MULTIPLAYER GAMES AND SINGLE-PLAYER GAMES

According to *"Overview of Multiplayer Mobile Game Design" (2003),* the following are the differences between multiplayer games and single-player games:

- Players provide the struggle
- Games must be repeatable
- Handling drops gracefully
- Short play session
- Game balance

2.2 MOBILE GAME CATEGORIES

Based on *"Planning a Game Application" (2003),* the current common mobile games categories include Arcade, Puzzle, Action, Sport, and Traditional. Arcade games are either based on classic arcade titles such as Asteroids, Pac Man, and Space Invaders, or are original titles with the same kind of game play. Since early arcade machines were quite limited in their media display capabilities and processing power, the style is an obvious one to translate to mobile devices.

Puzzle games are soloplay games that engage puzzle-solving skills, sometimes with continuous motion as in Tetris, sometimes more turn-based, in the manner of Bejeweled or Snood (popular PC puzzle games). These games tend to appeal more to casual gamers, rather than hardcore, which means that the potential market is enormous, but does not generally support high price points.

The term "Action game" is somewhat nebulous. Generally, it refers to a game in which a player controls a single character that moves through space, often engaging in combat with opponents. Player skill, rather than resource management or another type of challenge, is the focus of the game. Examples include Tomb Raider, Spyro the Dragon, and Sonic the Hedgehog.

Sports games are based on real-world (or sometimes imaginary) sports, and they constitute a large portion of the console and PC market, with titles like John Madden Football and FIFA Soccer. Given the small screen real-estate and limited control of mobile devices, it is hard to recreate team sports, but many early console sports titles faced similar issues, which were resolved by, for instance, offering hockey games with four team members per side. This approach will work for mobile games as well successful sports games are often based on licenses from sports personalities or sport associations.

Traditional games include board or card games such as Othello (also known as Reversi), Tic-tac-toe, Patience (Klondike Solitaire), Poker, and the like. Such games, as well as puzzle games, are generally well-understood and easy to design and program.

2.3 MULTIPLAYER GAME TYPES

A cellular network environment is fundamentally different from a fixed network environment, and this should always be kept in mind when developers are designing network games for a cellular network. Games should focus on the strengths of the medium while being designed around its weaknesses. For instance, currently multiplayer games over a cellular network cannot be very fast-paced because of the high latency involved, but that does not mean they cannot be appealing to users. The following list describes typical games that are generally suitable for cellular networks as in *"Multiplayer Game Performance over Cellular Network"(2004)*:

- **Multiplayer solo-play game:** Players feel as if they are playing in a multiplayer game, but each player is facing a single-player challenge with scores being compared at the end.
- **Round-robin game:** Each player takes his/her turn in order, and players wait until it is their turn to act.
- **Simultaneous movement game:** Also a turn-based game. Each player plans his/her moves independently from others, and sends the orders to the server. The server resolves the turn and dispatches the results to all of the players.
- **Act-whenever game:** The game persists for a long period of time, and players can sign in for short periods at a time to perform game actions.
- **Slow-update game:** The game operates continuously on the server, and players can sign in to change the behavior of their character(s).
- **Latency-adapted game:** In real-time games, the latency can be hidden behind the game rationale, for example, firing slow-moving missiles at enemy ships.

2.4 INFRARED AND ITS DRAWBACKS

Generally, according to Suvak (2000), infrared is used to provide wireless connectivity technologies for devices that would normally use cables for connectivity. Infrared is a point-to-point, narrow angle (30° cone), ad-hoc data transmission standard designed to operate over a distance of 0 to 1 meter and at speeds of 9600 bps to 16 Mbps.

Although infrared is regarded as the solution for cable connections between computers to network, printers and other devices, it has some drawbacks in multiplayer mobile gaming context. Furthermore, with another wireless connectivity solution, which is Bluetooth, infrared has become the second choice as the connection means for multiplayer mobile game sue to its limitations.

As mentioned by Franklin (2003), Infrared communications are fairly reliable and do not cost very much to build into a device, but there are a couple of drawbacks. First, infrared is a "line of sight" technology. For example, you have to point the remote control at the television or DVD player to make things happen. Furthermore, infrared signals cannot pass through walls or ceilings, meaning that at least one receiver and one transmitter is required in every room. Another drawback is that infrared is almost always a "one to one" technology. You can send data between your desktop computer and your laptop computer, but not your laptop computer and your PDA at the same time.

2.5 BLUETOOTH TECHNOLOGY

According to Franklin (2003) in his article *"How Bluetooth Works"*, Bluetooth is a specification for using low-power radio to link phones and computers over short distances without wires. Bluetooth networks use a dynamic topology called a piconet or PAN. Piconets contain a minimum of two and a maximum of eight Bluetooth peer devices. Bluetooth communicates on a frequency of 2.45 gigahertz, which has been set aside by international agreement for the use of industrial, scientific and medical devices (ISM). Bluetooth is intended to be a standard that works at two levels:

- It provides agreement at the physical level - Bluetooth is a radio frequency standard.
- It also provides agreement at the next level up, where products have to agree on when bits are sent, how many will be sent at a time and how the parties in a conversation can be sure that the message received is the same as the message sent.

Bluetooth is intended to get around the problems that come with both infrared and cable synchronizing systems. From the user's point of view, there are three important features to Bluetooth:

- **It is wireless.** When users travel, there is no worry about keeping track of a briefcase full of cables to attach all of the components, and office can be designed without wondering where all the wires will go.
- **It is inexpensive.**
- **Does not require user input.** Bluetooth does not require the user to do anything special to make it work. The devices find one another and strike up a conversation without any user input at all.

2.6 BLUETOOTH SPECIFICATION

The Bluetooth Specification as explained in *PaloWireless Bluetooth Resource Center* (2003) website describes how the technology works. It is also known as the Bluetooth protocol architecture. The following figure shows the protocol architecture of Bluetooth with brief explanation in Table 1:

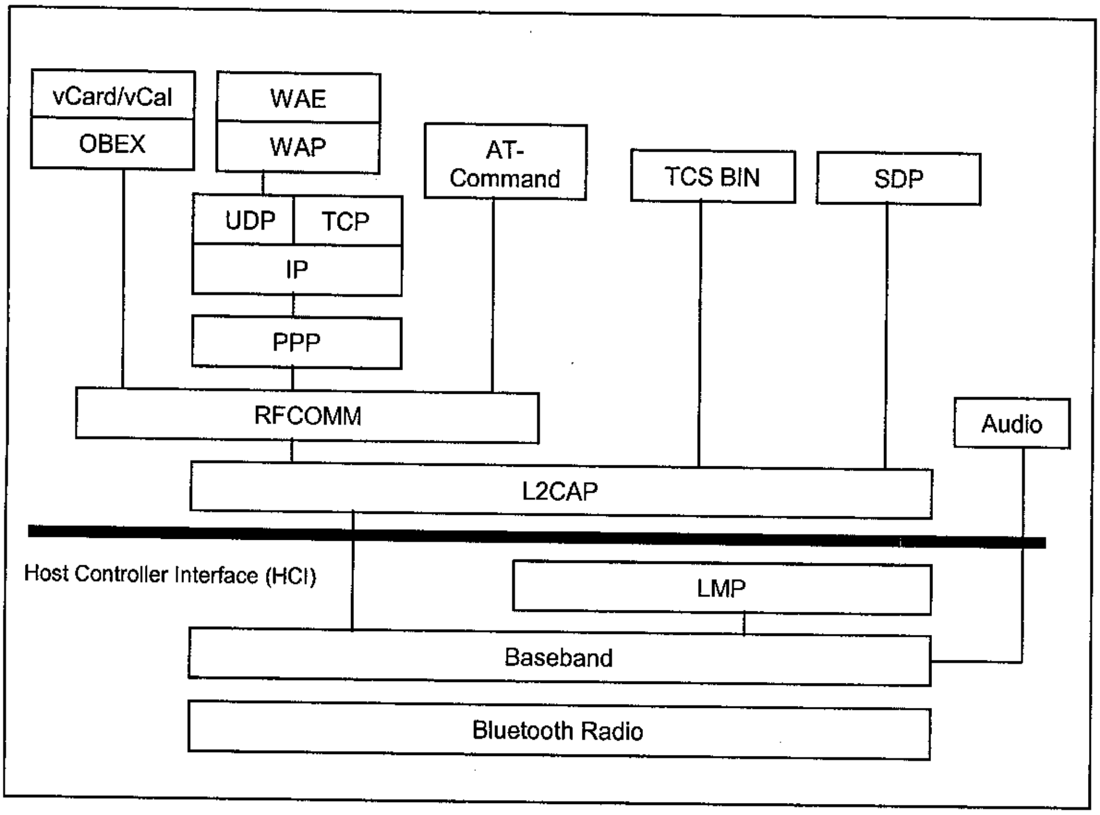


Figure 1: Bluetooth Specification Protocol Stack

Radio	v1.1	The **Radio** layer defines the requirements for a Bluetooth transceiver operating in the 2.4 GHz ISM band.
Baseband	v1.1	The **Baseband** layer describes the specification of the Bluetooth Link Controller (LC) which carries out the baseband protocols and other low-level link routines.
LMP	v1.1	The **Link Manager Protocol (LMP)** is used by the Link Managers (on either side) for link set-up and control.
HCI	v1.1	The **Host Controller Interface (HCI)** provides a command interface to the Baseband Link Controller and Link Manager, and access to hardware status and control registers.
L2CAP	v1.1	**Logical Link Control and Adaptation Protocol (L2CAP)** supports higher level protocol multiplexing, packet segmentation and reassembly, and the conveying of quality of service information.
RFCOMM	v1.1	The **RFCOMM** protocol provides emulation of serial ports over the L2CAP protocol. The protocol is based on the ETSI standard TS 07.10.
SDP	v1.1	The **Service Discovery Protocol (SDP)** provides a means for applications to discover which services are provided by or available through a Bluetooth device. It also allows applications to determine the characteristics of those available services.

Table 1: Bluetooth Specification Protocol's Components

2.7 BLUETOOTH CONNECTION SETUP FOR MOBILE GAME

According to "*Games over Bluetooth: Recommendations to Game Developers*" *(2003)*, Bluetooth devices need to exchange information before actual data transmission. This information exchange involves inquiry, service discovery, and remote name request.

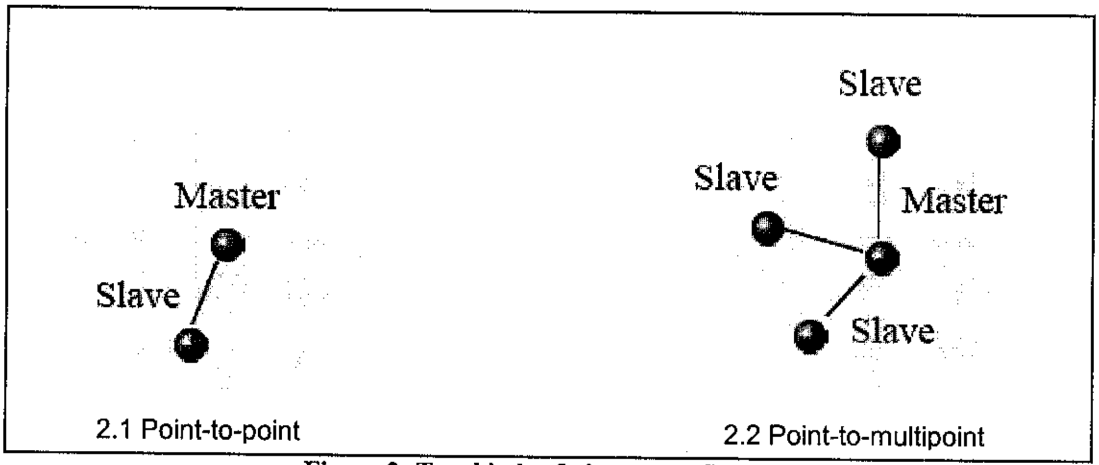
Since Bluetooth devices are mobile and form networks dynamically, they need a way to find other nearby devices to connect to. This process is called the inquiry procedure in the Bluetooth Baseband Specification. From the point of view of an application, inquiry means collecting Bluetooth addresses and determining the class of device (CoD) in the vicinity.

For inquiry, an access code is used. There is one General Inquiry Access Code (GIAC) to inquire for any nearby Bluetooth devices, and a number of Dedicated Inquiry Access Codes (DIAC) that inquire only for a specific type of device. The inquiry access codes are derived from reserved Bluetooth device addresses.

The next step is followed by service discovery, which is based on the Service Discovery Protocol (SDP). In wired networks, there is normally a central infrastructure for managing connections; Bluetooth does not provide this. Instead, connections are made dynamically, and devices must determine what services are provided on discovered devices. This is called service discovery, which is the process by which applications locate and gather information about other services in the vicinity. Inquiry just provides the user with bulky BD addresses. To obtain user-friendly names for remote devices, the remote name request procedure is used.

2.7.1 Piconet

A basic Bluetooth network is a piconet. The device that invites other devices into a piconet becomes a master, and a device that accepts such an invitation becomes a slave, although the roles can be switched later in some Bluetooth stacks. The master role does not imply any privileges, but means that the master device governs the baseband synchronization between devices. Figure 2 illustrates two kinds of piconet configurations, the first with a point-to-point connection (suitable for head-to-head games) and the second with point-to-multipoint connection (suitable for games with three to seven players). The slaves in the piconet only link to the master; there are no direct links between the slaves.



2.1 Point-to-point

2.2 Point-to-multipoint

Figure 2: Two kinds of piconet configurations

2.7.2 Host Role

Typically, in small-scale multiplayer games, all devices run the same software, and any can act as host. A host should perform the following actions:

1. Disable page scan and inquiry scan.
2. Register the service, which is unique for the game.
3. Start inquiry to obtain all Bluetooth devices with the appropriate CoD field in range. (The host/clients should perform an inquiry/inquiry scan on DIACs to reduce the number of responses.)
4. Perform a service search to find the devices that have started the game and to retrieve the player's name.
5. Present the found devices that have the game running to the host user.
6. Allow the host user to select multiple devices he or she wants to connect to.
7. Connect the devices the host has selected.

2.7.3 Client Role

On the other hand, the client should perform the following actions:

1. Register the service, which is unique for the game. Also create a player name as a service attribute. As default, use the Bluetooth device name.
2. Enable page scan and inquiry scan. (The host/clients should perform an inquiry/inquiry scan on DIACs to reduce the number of responses.)
3. Wait for incoming connection request.
4. Present to the user the choice to join or cancel the incoming connection request.
5. Set inquiry scan address back to GIAC.

2.8 SYMBIAN AND SYMBIAN OS

Symbian Ltd. is owned by Ericsson, Nokia, Panasonic, Psion, Siemens and Sony-Ericsson. Symbian delivers an advanced, open, standard operating system to its licensees. Symbian OS is flexible and scalable enough to be used in the variety of mobile phones to meet a wide range of user requirements.

According to *"Symbian on Bluetooth"(2001),* Symbian OS is a 32-bit multi-tasking operating system that is specifically designed for portable, battery-powered mobile phones. Some of the key features that make Symbian OS ideal for developing communications software include:

- A flexible client-server based architecture and programming systems to support compact but powerful software on wireless information devices.
- Instant access to user data using an effective multi-tasking kernel at the lowest level, and a very easy user interface framework at the highest level.
- Robust software design achieved through component-based object orientation and comprehensive use of design patterns and frameworks.
- Adaptability to different device types and different underlying communication technology.
- C++ and Java software development kits.

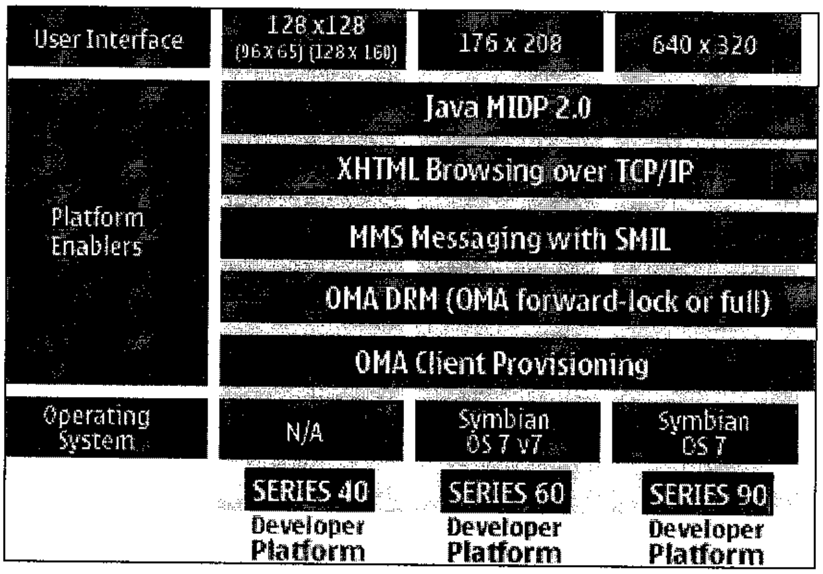
In *"Symbian OS Communications Programming"(2002),* Symbian OS is defined as a system with combination of several different elements that can be applied to several different designs of computing devices. There are six elements that combine to make up Symbian OS:

- the core operating system, commonly called a kernel
- a collection of middleware for system services
- a set of resource managers, called application engines
- a framework for designing user interfaces
- methods for synchronization with other machines
- a Java virtual machine implementation

These elements combine to produce software made to control devices from handheld computers to mobile phones.

2.9 DEVELOPER PLATFORM FOR SERIES 60 AND THE SERIES 60 PLATFORM

Nokia offers a range of Developer Platforms, namely Series 40, Series 60, and Series 90 that provide a consistent implementation of the leading mobile platforms. Figure 3 depicts the distinctive characteristics among those series, which is taken from the Nokia Forum website.



User Interface	128 x128 (96x65) (128x 160)	176 x 208	640 x 320
Platform Enablers	Java MIDP 2.0		
	XHTML Browsing over TCP/IP		
	MMS Messaging with SMIL		
	OMA DRM (OMA forward-lock or full)		
	OMA Client Provisioning		
Operating System	N/A	Symbian OS 7 v7	Symbian OS 7
	SERIES 40 Developer Platform	SERIES 60 Developer Platform	SERIES 90 Developer Platform

Figure 3: The characteristics for each developer platform

In *Developer Platform 2.0 for Series 60: Introductory White Paper* (2003), it states that the Developer Platform for Series 60 is a complete smartphone software package that provides a reliable base that can be customized for manufacturers' own hardware designs. The Developer Platform for Series 60 has been designed with standardization in mind while still maintaining a high degree of flexibility, and comes with a wide range of applications and communications capabilities.

The Developer Platform for Series 60 is a base set of technologies that are a mandatory part of any Series 60 device that is based on the particular version of the platform. The term "Series 60 Platform" encompasses the technologies of the relevant version of the Developer Platform for Series 60 but also provides an optional range of technologies that licensees may wish to support.

The Developer Platform for Series 60 is built on Symbian OS technology and provides a wide range of technology standards. For application development, C++ is the native programming language of the Developer Platform for Series 60, and public APIs are provided to allow developers to harness the full capabilities of the platform. The platform also includes Java APIs and an XHTML browsing environment via TCP/IP.

A number of devices currently on the market support the Developer Platform for Series 60. Nokia has released the Nokia 7650 and 3650 imaging phones as well as the Nokia N-Gage mobile game deck, all of which are based on Developer Platform 1.0 for Series 60. In addition, Nokia has also released Nokia 6600 imaging phone, based on Developer Platform 2.0 for Series 60 in the fourth quarter of 2003.

2.10 TECHNOLOGIES FOR GAME DEVELOPMENT

In *"Developer Platform 1.0 for Series 60: Conected Games"(2003),* it is stated that there are three main technologies for developing games for Developer Platform 1.0 for Series 60. The three main technologies are described in the following text.

2.10.1 Browsing

Browsing is a key enabler for various mobile services, including games. Browsing games are played by going to the game provider's URL, downloading one or more pages, entering text or choosing a menu option, submitting the data to a server, and then viewing more pages. This is a limited way to play games, mainly because all of the processing is done by a remote server rather than locally on the device — that is, all of the game play is done Over-the-Air (OTA), and is therefore subject to the problem of air network latency.

2.10.2 C++

Series 60 games can be written in C++ and installed onto a device using a .sis file. The Series 60 SDK offers a multitude of APIs for use by third-party C++ developers that enable the development of feature-rich high-performance games.

The communication technologies that could be used to develop a connected game in C++ are Serial Communications, Infrared Sockets, TCP (UDP)/IP Sockets and Bluetooth.

2.10.3 Java MIDP 1.0

Mobile Information Device Profile (MIDP) is one of the profiles defined by Java 2 Platform, Micro Edition (J2ME™), a version of the standard Java 2 Platform designed to run in an environment with constrained resources. Along with the Connected Limited Device Configuration (CLDC), MIDP provides Java APIs for the development of applications for mobile devices. Applications written using MIDP APIs are called MIDlets.

Games written for the Developer Platform 1.0 for Series 60 using MIDP can take advantage of APIs for sprite animation, connectivity via HTTP, and multithreading, among others. HTTP is the only communication technology supported by MIDP 1.0

2.11 EXAMPLES MULTIPLAYER MOBILE GAME DEVELOPMENT

There were various developments on mobile games and the products are commercially available online. However, those commercial products are mainly single player mobile games or 2-player mobile games. The following examples are some development of the multiplayer mobile games that are closely related to the project. However, there is no example of multiplayer mobile game via Bluetooth technology with more than two players that can be found.

2.11.1 Four-in-a-Line

Based on a review in My-Symbian.com website, Four-in-a-Line is a board game recreated in mobile format for Nokia 7650 and Nokia 3650. Four-in-a-line is suitable for beginners and experts alike. The aim of the game is very simple, which is to get four pieces in a line vertically, horizontally or diagonally. User can choose to play against the computer or challenge another Nokia 7650 or Nokia 3650 or SonyEricsson P800 mobile phone owner anywhere in the world via Short Messaging Service (SMS). The game allows user to play multiple SMS games with multiple different players at the same time. Figure 4 shows the interface design for the game.

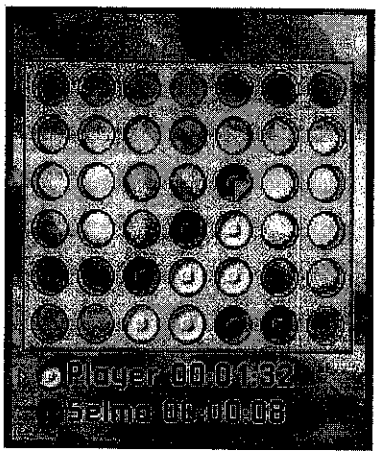


Figure 4: Four-in-a-Line

2.11.2 Agile Fighter Bluetooth

Another review in My-Symbian.com website features the Agile Fighter Bluetooth game. Agile Fighter is a fast paced kung fu game for Series 60 mobile phone featuring stunning graphics and exciting single-player and multiplayer gameplay. Players can be linked up over Bluetooth. The maximum number of player is two. When player is alone, the game can be played in single-player mode against the phone. Figure 5 depicts the interface of the game.

Figure 5: Agile Fighter Bluetooth

2.11.3 Constellation Othello

Constellation Othello is a game based on the popular reversi board games. The review of this game is obtained from the Noumena *Productions website(2002).* Constellation Othello user play as one of the many available game characters to challenge the evil Othello master. Two modes are available, which are single player mode versus the application and multiplayer mode where two players are connected via Bluetooth technology. Figure 6 shows the main screen of the game.

Figure 6: Main Screen of Constellation Othello

2.11.4 The Five

Another game from the *Noumena Productions website (2002)* is The Five. The Five is based on Wu-Zi, a classic chess game from ancient China. Also known as Renju in Japanese, the game is popular today around the world as a simplified version of GO that will challenge player's strategy thinking skills. The game can be played in single player mode and also in two-player mode using the Infrared or Bluetooth as the connection means. The following figure shows the interface of the game.

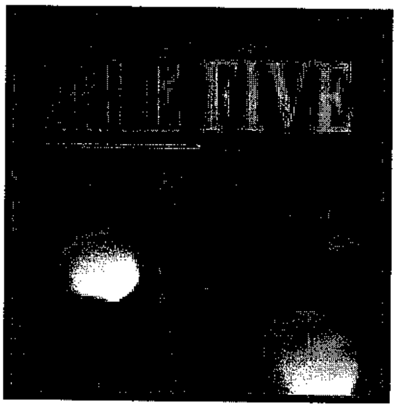


Figure 7: Main Screen of The Five

2.11.5 Mobile VR Pool

Mobile VR Pool is a full 3-Dimensional (3D) game developed using MGS. With Mobile VR Pool, players can easily rotate the table along any axis to aim and shoot. Mobile VR Pool also features fast animation and great sound. As similar with other game example, this game can be played in two modes, namely single-player mode and 2-player mode via Bluetooth. This game review is also obtained from *Noumena Productions website (2002).* Figure 8 shows the splash screen of the mobile game.

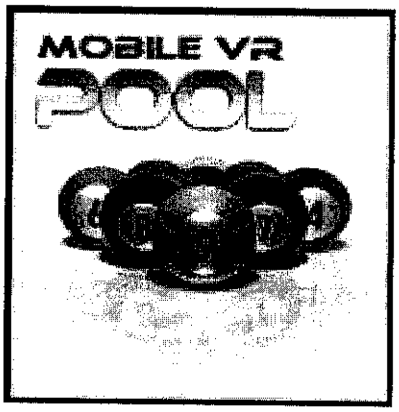


Figure 8: Splash Screen of Mobile VR Pool

CHAPTER 3
METHODOLOGY / PROJECT WORK

3.1 PROCEDURE IDENTIFICATION

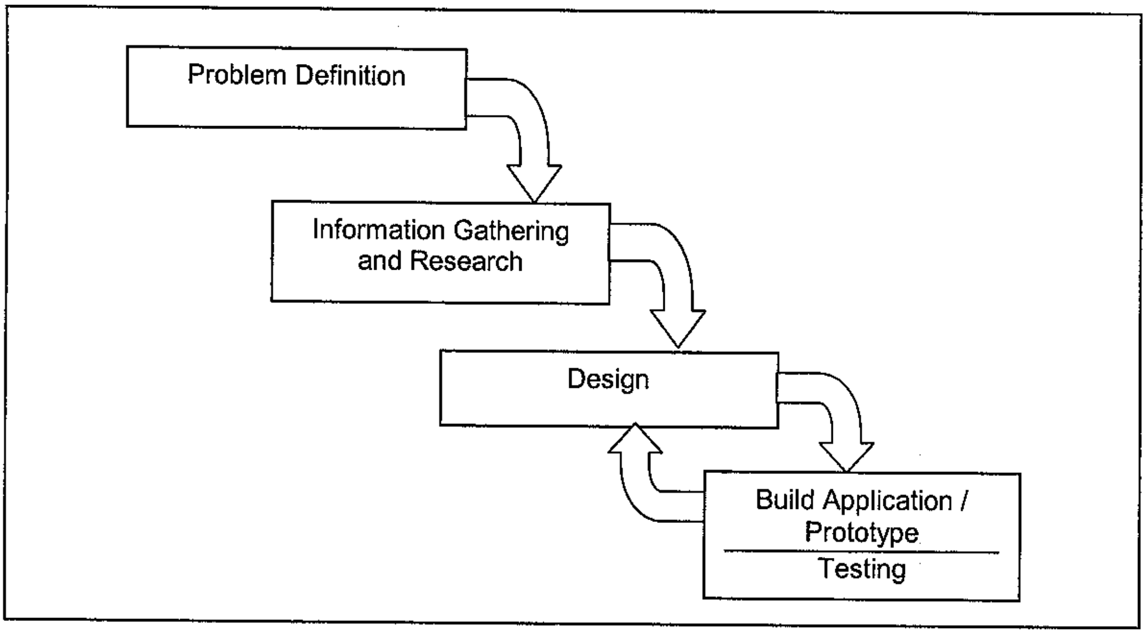

Problem Definition
Information Gathering and Research
Design
Build Application / Prototype
Testing

Figure 9: Diagram of the project methodology

The methodology used in this project is the combination of waterfall and prototyping model as shown in Figure 9. The following will explain in detail on each process in the prototyping methodology.

3.1.1 Problem Definition

The project begins by identifying the broad problem area and followed by defining the problem area in detail. The project basically looks into the 3-player mobile game development using the point-to-multipoint Bluetooth connection.

3.1.2 Information Gathering and Research

Next, information gathering process will be performed to collect all the necessary resources. At this stage, the project will basically look into the research of multiplayer (specifically three-player) mobile game with Bluetooth technology as the connection means for the mobile phone devices. A thorough literature review is done through journals, reference books and internet to further understand the step-by-step development procedure of multiplayer mobile game and to learn the programming language used to develop the application. Besides that, information on Bluetooth technology is gathered to do reviews on its drawbacks. One of the major resources obtained is from the Forum Nokia website, which contains various references on mobile game development and Bluetooth technology.

3.1.3 Design

The design process involves the game design steps in which will focus on the preparation of Game Design Specification. It specifies the game flow, user interface (UI), fundamental gameplay algorithm, and what media assets need to be created for the game. The media assets refer to the media used in the game such as graphics. The Game Design Specification should be as complete as possible, but unlike more conventional software development specifications, it can and should be revised frequently during the game development process.

3.1.4 Build Application / Prototype and Testing

Based on the findings, the project aims to develop a functional multiplayer mobile game application by the end of the project. The application is developed using Microsoft Visual C++ 6.0, which acts as the Integrated Development Environment (IDE) for Symbian OS SDK. The application developed in a personal computer is then converted to Symbian Installation System (.sis) file to enable the application installation on the mobile phone device. Prior to that, the development of the mobile game is coded and debugged in an emulator.

Next, play testing is performed, which is a significant step in game development. Play testing is not testing for bugs (although play testers should note and log bugs), but testing for fun. The game can be iteratively refined as a result of what the play testers report to refine the UI, provide additional useful features, drop irritating features, and improve game play in general.

3.2 TOOLS REQUIRED

3.2.1 Hardware

- Personal Computer with minimum processing speed of 750 MHz, 128 MB RAM and hard disk storage 1GB
- Mobile phones with Bluetooth technology (*The phone model owned is Nokia 3650 and will be used for demonstration purpose)*

3.2.2 Software

- Platform used is Windows 2000 Professional
- Proposed application/prototype to be developed in Microsoft Visual C++ 6.0
- Symbian 6.1 SDK v1.2
- PC Suite for Nokia 3650
- Java Runtime Environment 1.3.1
- Active Perl 5.6.1
- Adobe Photoshop 7.0

CHAPTER 4
RESULTS AND DISCUSSION

4.1 MOBILE GAME DEVELOPMENT TOOLS

Before the information gathering and research began, it is thought that the only programming language used to develop new version of mobile games is Java (using J2ME). However, after thorough literature review on multiplayer mobile game technology, it is discovered that multiplayer mobile game can also be developed using C++ for mobile phones that operate on Symbian OS. The communication technologies that can be used to develop connected games using C++ are Serial Communications, Infrared Sockets, TCP(UDP)/IP Sockets and Bluetooth. On the other hand, HTTP is the only communication technology supported by Java MIDP 1.0.

4.2 COMPONENTS OF SERIES 60 EMULATOR

The components of the Series 60 emulator are shown in Figure 10. It is significant to know the name of each components of the emulator to illustrate the mobile game application and the terms that will be described in the following sections.

The most frequent components that will be mentioned are left soft key, right soft key, navigation key and control pane.

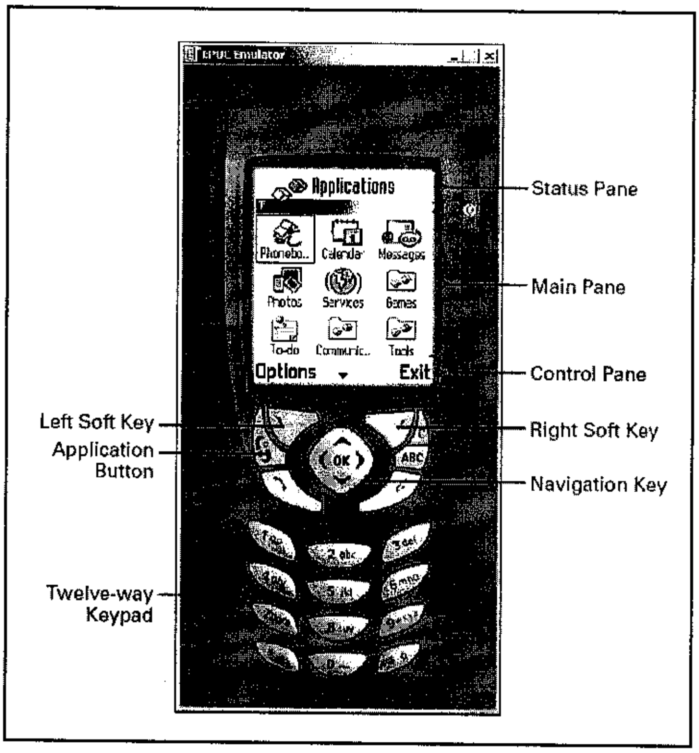


Figure 10: Series 60 emulator

4.3 GAME TITLE AND CONCEPT

The title of the 3-player mobile game developed is 'Odd1Wins' and uses Bluetooth as the connection means. The application is based on a conventional childhood style to determine the winner based on hand flipping. The one with the odd side is considered as the winner. Figure 11 is shown to further illustrate the concept of the game.

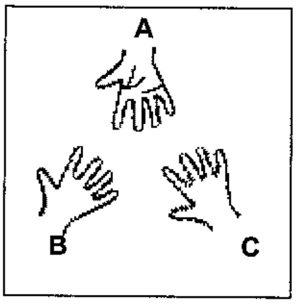


Figure 11: The concept of 'Odd1Wins'

Based on Figure 11, Player A is the winner since the other two players display the same side of their hand. In 'Odd1Wins', the first player who have won five times will be the final winner of the game session . The players' choice is made by pressing the left or right navigation key on the mobile phone and the decision time is controlled by a timer.

4.4 GAME DESIGN SPECIFICATION

In a mobile game design specification, it includes the documentation of the game flow, user interface (UI) design, fundamental gameplay algorithm and media assets to be included into the game. Each item of the game specification is described in detail as in the following section.

4.4.1 Game Flow

Figure 12 depicts the flow of the 'Odd1Wins' game application. The game starts when user selects the 'Odd1Wins' application icon on the appications main menu. Once the user has selected the application, a splash screen will appear showing the game title. On the same screen, user is required to press on the left soft key ('Options') or the right soft key ('Exit').

When user selects the right soft key, the application will be terminated and will return to the applications main menu. On the other hand, when left soft key is pressed, there are four menu items that can be chosen by the user as shown in Table 2.

Menu Item	Function
Introduction	An introduction document in text form will be displayed to show the concept of the game and requirements to play the game.
New Game	Two sub menu items will be shown, namely 'Server' and 'Client': **Server** - To initiate the game session **Client** - To get connected to the master
Help	A help document will be displayed which tells how to play the game.
Exit	To exit the game application.

Table 2: Menu items of the 'Options' menu

As stated in Table 2, two sub menu items which are 'Server' and 'Client' will be displayed when user selects New Game. One of the three players who want to play 'Odd1Wins' would have to initiate the bluetooth discovery and act as the Server (Master). The other two players will have to select 'Client' and act as Slave to the Bluetooth connection.

After selection is made, the Bluetooth devices will need to exchange information before actual data transmission. It includes device inquiry, service discovery and remote name request. When the connection is setup between the three devices, the game will begin.

The screen will first display the position of the players' virtual hands. A popup screen will request the players to enter their selection of hand flipping side. The decision time is limited and controlled by a timer. When the timer ends, the hand selection will be displayed as well as the winner's name for that round. If three players chose the same side of hands, a draw result will be given. The game will loop until one of the players score five winning times. Next, the final winner will be determined. The game session will end after the final winner result is displayed.

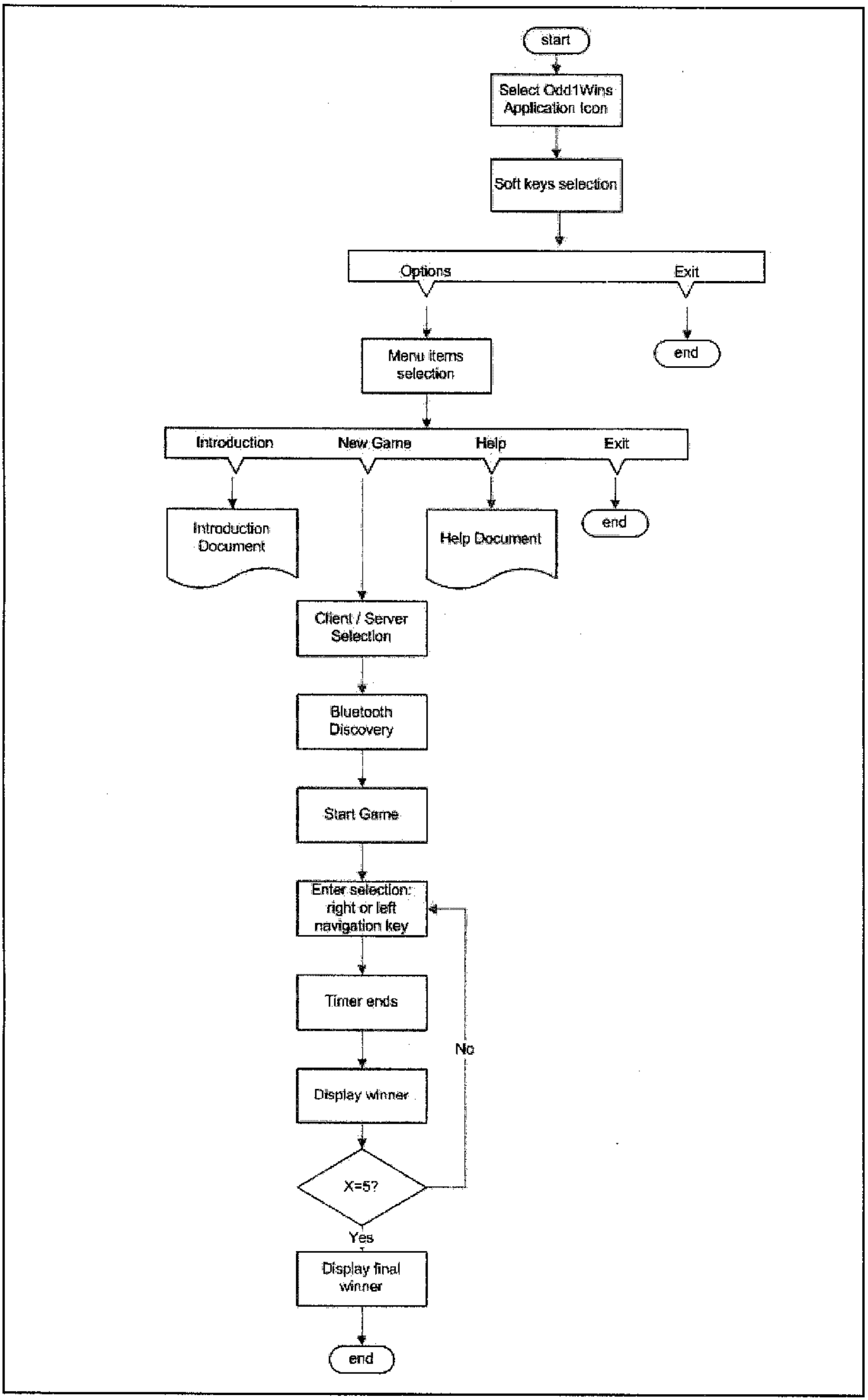


Figure 12: The Game Flow of 'Odd1Wins'

4.4.2 User Interface (UI)

The UI for the mobile game application is drawn to decide the elements to be included in the screen display. Figure 13 shows the UI designs that have been scratched manually during design procedure.

Based on Figure 13, Screen 13.1 shows the user interface that will be displayed on the mobile phone screen when the game application icon is selected in applications main menu. In Screen 13.1, the application icon and the title of the game will be shown in the Status Pane. The splash screen is displayed in the Main Pane. User will have to press on the left or right soft key to continue with the application.

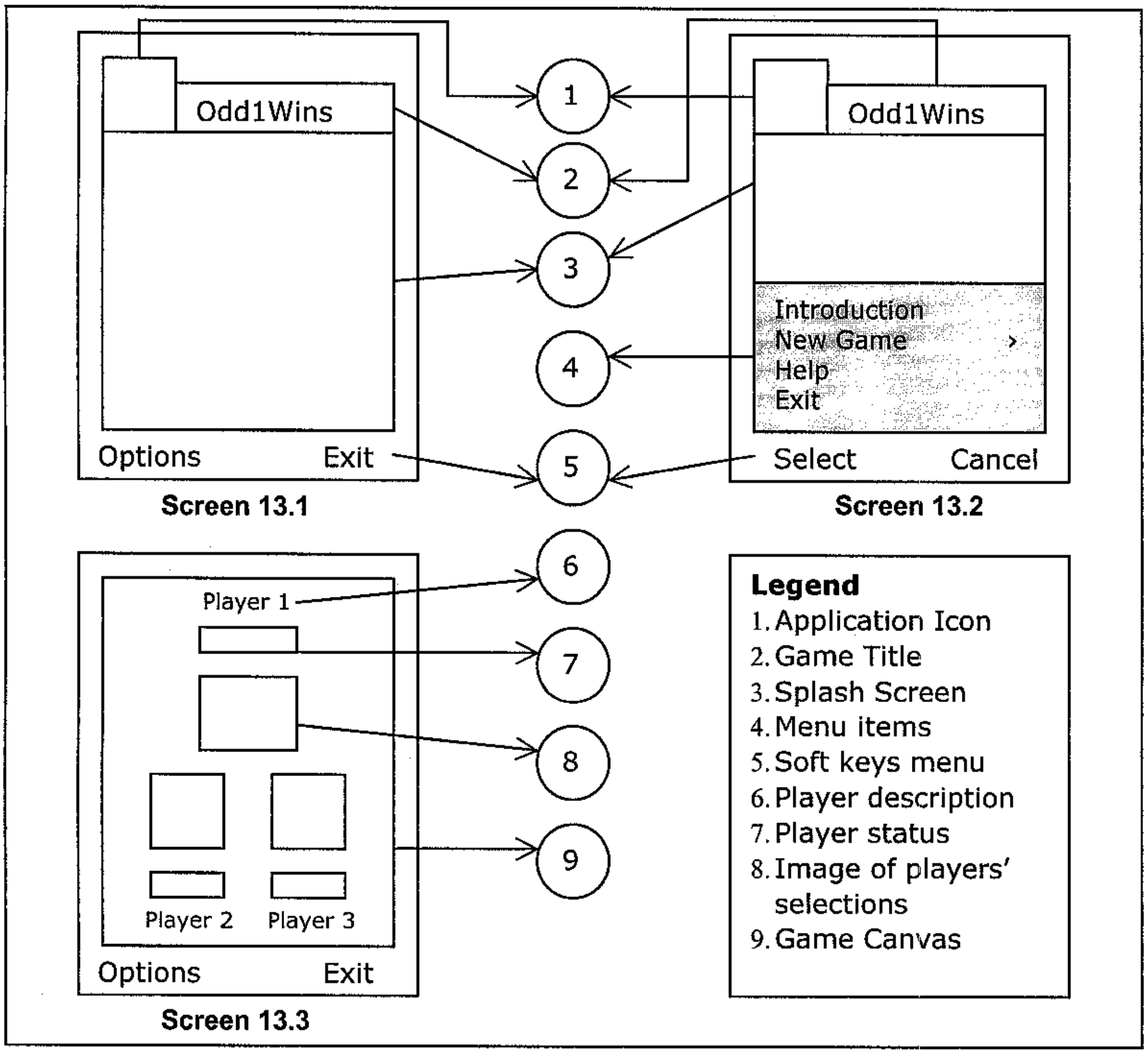


Figure 13: The user interface design of 'Odd1Wins'

Screen 13.2 illustrates the user interface when the 'Options' menu is selected. The menu items are displayed which include 'Introduction', 'New Game', 'Help' and 'Exit'. If there exists submenu item, there will be an arrow indicating that the menu item can be expanded as in 'New Game'.

Screen 13.3 depicts the user interface of the players' position during the gaming session. There will be a label indicating the players' device name, a status bar to indicate the winning frequency and also the image of the player's virtual hand. The game convas will cover the status pane and also the main pane.

4.4.3 Fundamental Gameplay Algorithm

In general, a gameplay algorithm is any set of rules or calculations that affect response to player input or the behavior of objects or characters in the game. In the case of 'Odd1Wins', the only important algorithm is the way to determine the final winner of the particular game session. The final winner is determined by the first player who scores five winning times during the session. Thus, during every round, there will be an algorithm to check the amount of winning time for each player. When the winning time of one of the players reaches five, the game will end and final winner will be announced.

4.4.4 Media Assets

There are various graphics that need to be created for the game application. The graphics include the application icon, title screen (splash screen), players' virtual hands, and status bar. All the graphics are produced using image editting software, Adobe Photoshop 7.0.

4.5 GAME DEVELOPMENT

The mobile game application is developed based on the Game Design Specification produced earlier. The game is coded in Microsoft Visual C++. The following figures show the snapshots of the mobile game application accompanied by explanation. The screens are captured when the application is tested on the EPOC emulator.

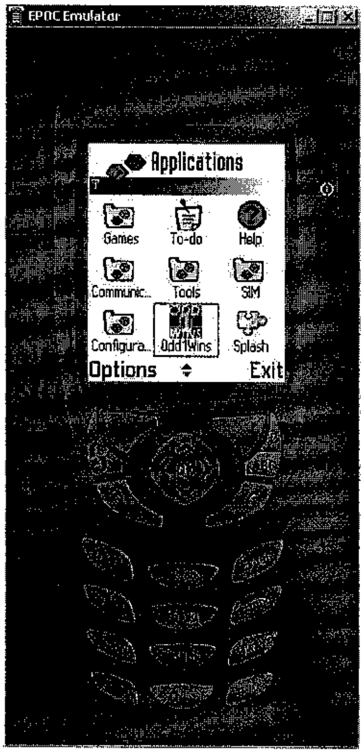


Figure 14: Applications Main Menu

Figure 14 shows the interface when the emulator is started. It displays all the applications contained in the emulator. The 'Odd1Wins' icon resides in the main menu as one of the application. To access to the game application, user will navigate to the 'Odd1Wins'icon using the navigation key pad.

The game's splash screen will be displayed as shown in Figure 15 when 'Odd1Wins' icon is selected. It shows the title screen of the application and user will need to press the soft key to continue. The right soft key is labeled as 'Exit' while the left soft key contains the 'Options' menu.

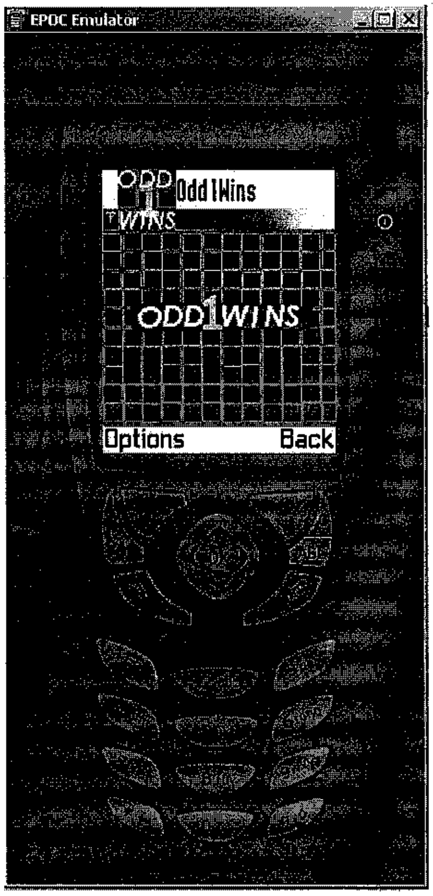


Figure 15: Splash Screen of 'Odd1Wins'

Figure 16 shows the menu items for the 'Options' menu. There are four menu items, which are 'Introduction', 'New Game', 'Help', and 'Exit'. The functions of these menu items have been explained in the earlier section. The application user interface has been designed and developed based on the Game Design Specification.

When user selects the 'Introduction' menu item, an introduction document will be displayed as in Figure 17. The document basically introduces the game to users and tells the requirements of the game.

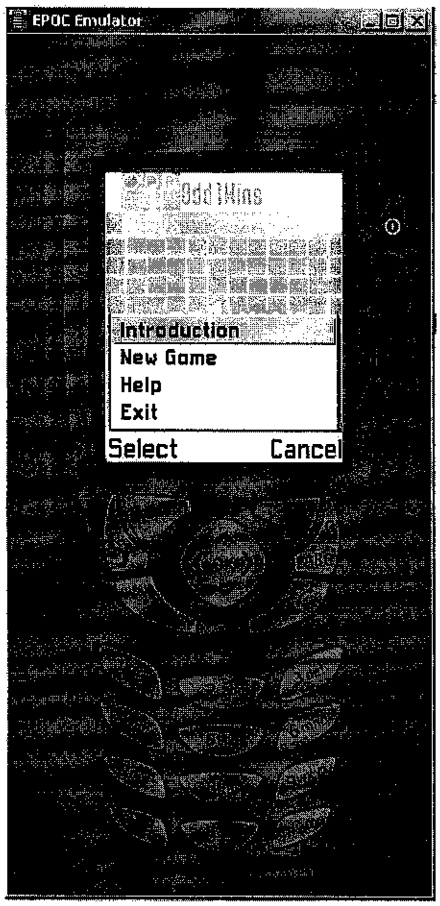
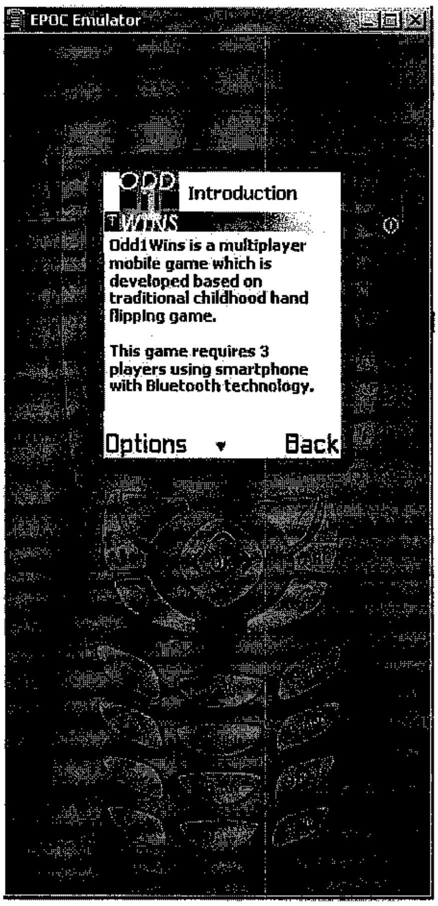


Figure 16: Menu items of the 'Options' menu



Odd1Wins is a multiplayer mobile game which is developed based on traditional childhood hand flipping game.

This game requires 3 players using smartphone with Bluetooth technology.

Figure 17: Introduction Document

On the other hand, Figure 18 shows the interface design when 'New Game' is selected. User will be prompt to select either Client or Server. When the Bluetooth connection is established, a screen with the players' position on the emulator will be shown as in Figure 19. Next, user will be asked to enter the hand side selection on that is illustrated in Figure 20. After selection is entered, the result is displayed on the screen and the player with odd selection will get a winning point. The procedure is shown in Figure 21.

If user needs help on the game, a 'Help' menu is provided. When it is selected, a help document will be displayed to teach user on how to play the game. The interface is shown in Figure 22.

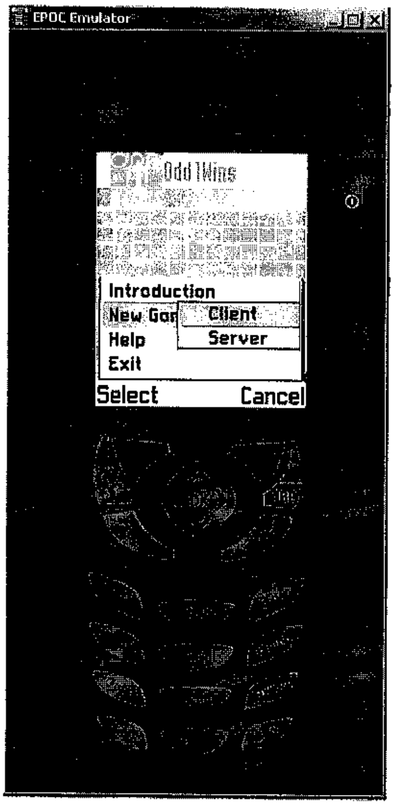


Figure 18: New Game's sub menu item

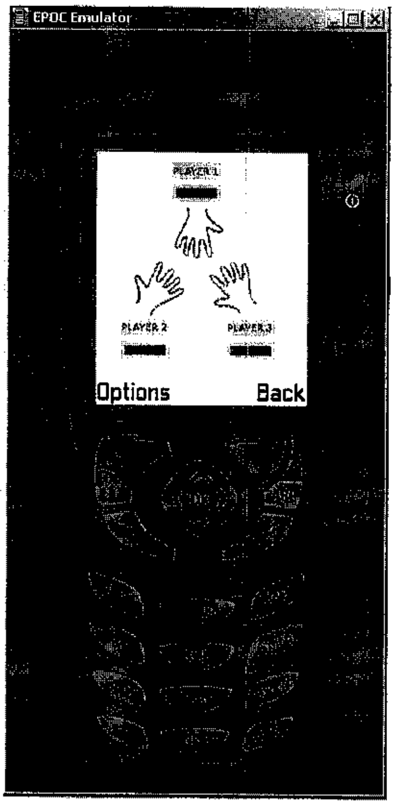


Figure 19: Players' Hands Position

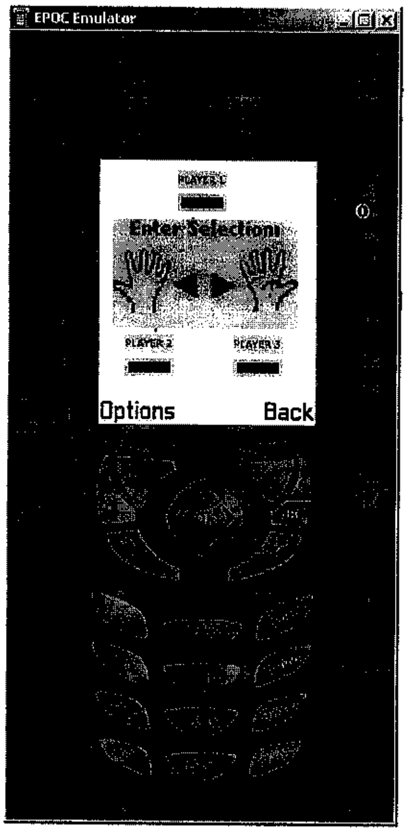


Figure 20: Selection screen

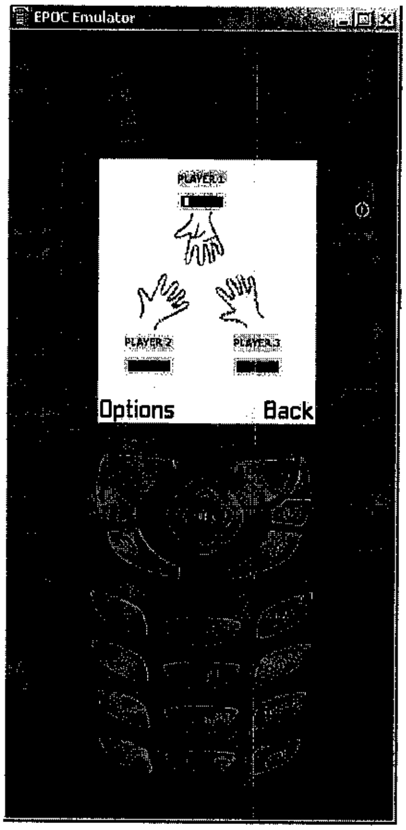
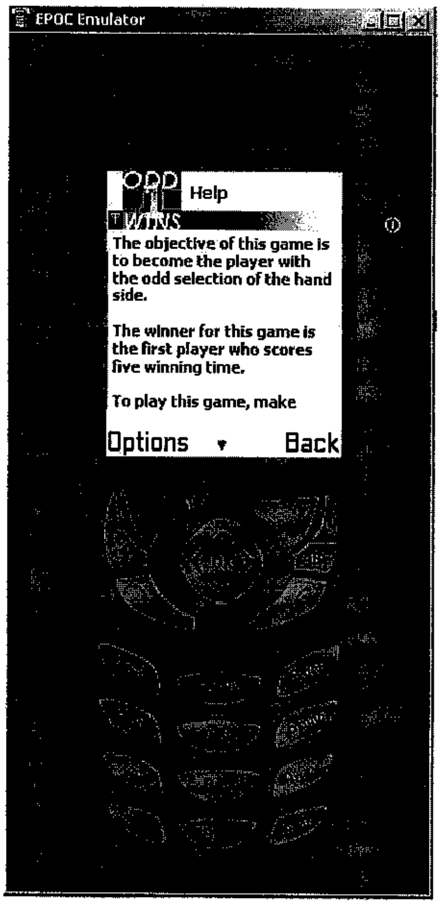


Figure 21: Selection entered by user



The objective of this game is to become the player with the odd selection of the hand side.

The winner for this game is the first player who scores five winning time.

To play this game, make

Figure 22: Help document

4.6 KEY DESIGN ISSUE AND CHALLENGES

There are various design issues and challenges faced during the development of this multiplayer mobile game. These issues and challenges are in terms of the phone's physical appearance, memory storage, and the application itself. These main issues are highlighted here.

4.6.1 Screen Size and Format

Mobile devices have screens that are much smaller than those on Personal Computer (PC) or consoles. Thus developer needs to take this limitation into consideration while designing the graphics, images and font sizes of a game application.

It is significant to ensure that all medias can fit into the screen area. Besides that, the screen size will determine the appropriate resolution for all the graphics.

4.6.2 Keypad Layout

Since the mobile phone used in this project is Nokia 3650, it is essential to note that the model has an unusual keypad layout as compared to other models. The keypad layout for the phone is shown in Figure 23.

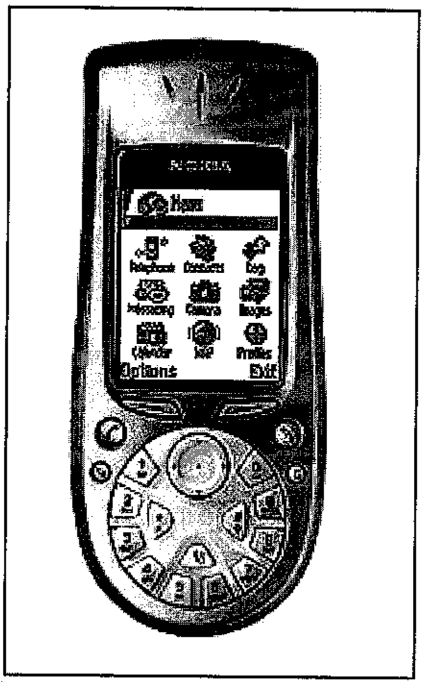


Figure 23: Keypad layout of a Nokia 3650

To ease the gaming session, only soft keys and navigations keys are used in developing the 'Odd1Wins' application. This will reduce users' unintuitive on the keypad layout difference, specifically on the numerical keypad.

4.6.3 Limited Application Size

Developers need to ensure that the application developed is in small size and does not exceed the limit. For an application developed for Series 60 phones, a file size of 100kb to 200kb is reasonable and preferably smaller than the size mentioned.

4.7 BLUETOOTH LIMITATION IN MOBILE GAME

The main feature of a Bluetooth is it does not need to be in line-of-sight mode to get connected to other devices. This has really increased the effectiveness and easiness in playing a multiplayer mobile game as compared to Infrared connection. Adding on to that, Bluetooth technology has made playing a multiplayer mobile game with more than two players possible.

In terms of comparing Bluetooth technology and infrared, it is obviously the former would be a better choice for a multiplayer mobile game application. The only limitation of Bluetooth is the 10 meters range distance between those connected devices. Thus, users who want to play the multiplayer mobile game application, they need to be within 10 meters with the other players, but line-of-sight mode is not required. So, the distance limitation would not really interrupt the gaming session unless user is thinking of connecting players from a far distance. If that were the case, user would have to opt for multiplayer mobile game, which uses other connection means such as SMS or WAP.

4.8 LIMITATIONS OF THE APPLICATION

There are few limitations on the application produced. All the limitations are described as below.

4.8.1 Clients Have to Enter Selection First

During the game session, the players who acted as the clients have to enter the hand selection before the server (master). This is because the clients' selection will be sent to master first before enabling the server to enter the selection before processing the result.

4.8.2 No Game Replay

The application does not support game replay. Each time the game is played, it can only cater for one session. Users have to reestablish the Bluetooth connection if they want to play another session.

4.8.3 No Animation and Sound

Mobile Game on Nokia Series 60 can have various advanced features such as animation and audio. However, due to time constraint, these features were not included in the application.

4.8.4 Fixed Number of Player

The application has fixed number of player, which is three. The game cannot be played by less than three or more than three players. The application needs the existence of three devices in order to start the game.

CHAPTER 5
CONCLUSION AND RECOMMENDATIONS

This project has highlighted the capability of Bluetooth technology in enhancing the multiplayer mobile game experience. Using Bluetooth as the connection means for a mobile game application enables more than two players play a mobile game simultaneously instead of two players if infrared is being used. Moreover, players do not need to be in line-of-sight mode during the entire game session.

Besides that, if a Bluetooth multiplayer mobile game is compared to other types of multiplayer mobile game besides using infrared, such as Short Messaging Service (SMS) and browsing, Bluetooth has a main advantage- it is costless.

The prototype demonstration shows the possibility of developing a mobile game application for more than two players. This can greatly utilize the capability of Bluetooth technology in mobile game industry.

Based on the limitation of the application developed, there are few recommendations that would like to be highlighted for future enhancement. The application can be improved by implementing simultaneous hand selection by master and slaves. Another enhancement that can be done is to enable users to replay the game without the need to reestablish the Bluetooth connection each time users would like to restart the game session. Besides that, the end-product can be enhanced by implementing audio instructing users to enter their selection, more appealing graphics and animation such as moving images to show the hand flipping.

On the other hand, the research can be further extended to study on multiplayer mobile game with more than three players as Bluetooth technology can support up to eight devices connected simultaneously.

REFERENCES

[1] "Overview of Multiplayer Mobile game Design" Dec 17, 2003. Jan 26, 2004 <http://www.forum.nokia.com>

[2] "Planning A Game Application" July 2003. Jan 26, 2004 <http://www.forum.nokia.com>

[3] Multiplayer Game Performance over Cellular Network, Jan 2004. Jan 31, 2004 <http://www.forum.nokia.com>

[4] Franklin, Curt "How Bluetooth Works" No date. Jan 26, 2004 <http://compnetworking.about.com/gi/dynamic/offsite.htm?site=http%3A%2F%2Fwww.howstuffworks.com%2Fbluetooth.htm>

[5] "Symbion on Bluetooth" Oct 2001. Feb 27, 2004. <http://www.symbian.com/technology/standard-blue.html>

[6] Jipping, Michael. *Symbian OS Communications Programming.* England : John Wiley & Sons, 2002

[7] "Bluetooth Tutorial – Specifications" 2003. Feb 25, 2004. <http://www.palowireless.com/infotooth/tutorial.asp>

[8] "Developer Platform 2.0 for Series 60: Introductory White Paper" August 2003. Feb 26, 2004. <http://www.forum.nokia.com>

[9] Developer Platform 1.0 for Series 60: Conected Games" Nov 2003. Feb 28, 2004. <http://www.forum.nokia.com>

[10] "Multi-Player MIDP Game Programming" Oct 29, 2003. Jan 27, 2003 <http://www.forum.nokia.com>

[11] "Noumena Productions" 2002. April 1, 2004 <http://www.noumenaproductions.com/>

[12] "Games Over Bluetooth: Recommendations to Game Developers" Nov 13, 2003. Jan 26, 2003 <http://www.forum.nokia.com>

[13] "Introduction to Mobile Game Development" Jan 2003. Jan 26, 2003 <http://www.forum.nokia.com>

[14] "Dialed In: MIDP 2.0 Sets the Standard for Mobile Gaming" No date. Jan 27, 2003 <http://www.sun.com/br/comms_421/feature_gaming.html>

[15] Bedford, St. Martin's "Using MLA Style to Cite and Document Sources" No date. Jan 30, 2003 < http://www.bedfordstmartins.com/online/cite5.html#1>

[16] Mitchell, Bradley "Wireless/Networking: Computer Networking Definition for Bluetooth" No date. Jan 25, 2003 <http://compnetworking.about.com/library/glossary/bldef-bluetooth.htm>

[17] McDonald "Wireless Networking: Does Infrared Have a Chance?"Aug 30 2001. Feb 29, 2004 <http://wireless.newsfactor.com/perl/story/13228.html>

[18] Akalugwu Bugnevicius, Evers, Pourtaran "Bluetooth vs Infrared – A Comparison" Nov 2003. March 1, 2004. <http://www.faculty.iu-bremen.de/birk/lectures/PC101-2003/17bluetooth/bluetooth/intro.html>

[19] Jode, Turfus "Symbian OS System Definition" Aug 2003. March 1, 2004 <http://www.symbian.com/developer/techlib/papers/SymbOS_def/2003-08-05_SyOSDef.pdf>