Personalised Mobile Picture Puzzle

by

Nur Syazwani bt. Rusdi

supervised by

Mr. Saipunidzam Mahamad

Dissertation submitted in partial fulfillment of

the requirements for the

Bachelor of Technology (Hons)

(Information Communication Technology)

JUNE 2006

Universiti Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

t
TK
5103.2
. N974
2006

1) wireless communication system

CERTIFICATION OF APPROVAL

Personalised Mobile Picture Puzzle

by

Nur Syazwani bt. Rusdi

A project dissertation submitted to the
Information Communication Technology Programme
Universiti Teknologi PETRONAS
in partial fufilment of the requirement for the
BACHELOR OF TECHNOLOGY (Hons)
(INFORMATION COMMUNICATION TECHNOLOGY)

Approved by,

_____

(Mr. Saipunidzam Mahamad)

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK
June 2006

ii

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.

NUR SYAZWANI BT. RUSDI

ABSTRACT

Pictures taken using camera phone are usually available for printing, sending to others via e-mails or MMS and if the pictures are copied to computer, the pictures can be modified using special software for image development. Therefore, a game that called Picture Puzzle will take the advantage of the pictures taken using camera phones in order to multiuse those pictures. The main objective of project is about developing a mobile game application in order to multiuse the picture taken using camera phone. The project had involved research and study on Mobile API, RMS and TiledLayer class from Game API with the purpose of developing the mobile game application. The project had consisted five main phases which are Planning, Analysis, Design, Development and Implementation. The Picture Puzzle will be displayed the picture taken as a scrambled grid and then the user will have to unscramble the picture puzzle. When the user manages to unscramble the puzzle, the game is completed. As conclusion, the project had able to achieve its objectives and the game application will be able to run as expected when it is implemented on the handphone.

ACKNOWLEDGEMENT

Alhamdulillah, after almost 1 year doing this Final Year Project, the project and Final Report are finally done. There were many people involved during the development of the project, which has contributed immeasurable amount of guidance, advice and assistance. My heartfelt thanks go to all of them for their commitment and supports in realizing the goals and the successful completion of the project.

First and foremost, I express my gratitude to our Merciful God, Allah S.W.T with His blessings for me to complete the Final Year Project and this Final Report.

I want to point my special thanks to Mr. Saipunidzam Mahamad who had supervised me throughout my project development period. I really appreciate his kindness and time to guide me to get through the project development. As a supervisor, he is really concerned and gave me the guidance for the sake of my knowledge and experiences in completing this project.

I also want to express my utmost gratitude to Siti Nabilah Ibrahim for her willingness in giving the full cooperation in order to complete the project. All your guidance and teachings in understanding the project needs were truly important in ensuring the successful completion of the project.

These special thanks go to my parents and family for all the helping hands and priceless advises. Thank you also for all the support and understanding in the development of this Final Year Project. All your guidance and support were truly important in ensuring the successful completion of the project.

Last but not least, these million thanks go to all my friends and colleagues especially Sharifah Maizatul Akma Syed Azimuddin, Siti Nuraisyah Jamaludin, and Noor Azira Mohd Anual, thank you for all the support and understanding during the completion of the project.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATION

- J2ME – Java 2 Micro Edition
- KVM – K-Virtual Machine
- CLDC – Connected Limited Device Configuration
- MIDP – Mobile Information Device Profile
- IMP – Information Module Profile
- RMS – Record Management System
- API – Application Programming Interface

CHAPTER 1

INTRODUCTION

1.1 Background of Study

Picture puzzle which is more popular with the name of jigsaw puzzle is a game that requires the player to assemble numerous small pieces of picture, usually oddly-shaped in order to produce a complete picture. The original picture puzzle game is so famous since it is believed that most of the people in world must have played the game. Therefore, the author had developed a game application based on the idea from the original picture puzzle game. The game application is a mobile application that had been implemented on a handphone. The game is called Personalized Mobile Picture Puzzle which is suitable to be shipped on Nokia Series 60 phones. Usually the image taken using camera phone can only be printed, save and set as phone's wallpaper. So, this newly developed game has used the image taken by the camera phone in order to create a puzzle that can be played by the user. The main concern of the project is to deliver a game application that uses Mobile Media API, RMS and TiledLayer class from Game API. The Mobile Media API is used for capturing photo using a camera phone. RMS is for save, load and delete the persistent records and makes use the TiledLayer class from Game API which consists of a grid of cells each of which can be filled with an image tile. The image taken will automatically displays as a scrambled grid. The user has to unscramble the image by re-arranging the tiles to complete the game.

1.2 Problem Statement

Mobile phone manufacturers are always adding new features to their phones, and one of the new features is a built-in camera. This new feature is very useful and

beneficial to the users because users carry their mobile phone with them everywhere and having a built-in camera on the handphone will allows the users to capture both planned and unplanned events taking place around them. The pictures taken can be printed like the pictures taken using the digital camera and also share the pictures with others. The users can share the pictures by sending MMS, e-mail and etc. Besides that, the pictures can be copied to the computer for other usages of those pictures.

Eventhough there are variety of usage of pictures taken using camera phone, but the pictures are not utilized widely. As mentioned earlier, the pictures taken are normally used as original picture not something that is modified. Thus, in order to fully utilize the picture taken, one way is by creating a game application from the picture. This game application will modify the picture taken and this will be an addition to the usage of picture taken using camera phone and allows the user enjoy more with the pictures they have taken. Beside that, the current picture puzzle game is too easy. It is because the current puzzle applies grid of size 4 x 4 only and the game also uses pictures that are permanently stores in the game application. So, in this project, the author wants to enhance the current game by adding a few new functions to the game application.

1.3   Objectives and Scope of Study

1.3.1   Objectives

1. To develop a mobile game application in order to make use of the pictures taken using the camera phone.
2. To apply J2ME and Symbian OS technology.
3. To embed new game option on handphone so that the users can enjoy with their handphone features.

1.3.2 Scope of Study

In order to develop the Picture Puzzle game application, there are a few scope of study that needs to be followed. Here are the scopes of study:

- Research based on programming language and operating system for mobile which are J2ME and Symbian OS in order to get better under standing about mobile application.

- Research based on game application development regarding the important development processes.

- Design the main elements of the application – user interface and UML class diagram

- Develop a prototype application – test the prototype in order to improve further before it is implemented.

- Implement the application and deliver the final product.

1.4 Relevancy and Feasibility Study

1.4.1 Relevancy

This project is relevant to the course taken as an IT student. The project had used the skills that have been learned during the studies especially the programming skills which are related to JAVA. Eventhough the project is applying J2ME but it just almost the same like normal JAVA programming that had been learned during JAVA class but a few new things need to be learned and studied for better understanding. For the project documentation, the Research and Computer Project Management classes had helped a lot. Most of the knowledge that had been learned during both classes had been used and applied in order to complete the required documentation for the project. Moreover, the project is very relevant to the studies and it actually had given the opportunity to apply the skills and knowledge that have been develop during the studies and it allows to gain new knowledge and abilities that will be useful and beneficial in the future undertaking.

1.4.2 Feasibility Study

The project had been developed within the given time duration. The time duration is very flexible and enough to produce a good end product. Furthermore, the skills and knowledge that had been used for the project are available and can be applied in order to develop the project successfully. There are new skills and knowledge required but they had been obtained by doing the study and research on the related subject. For tools that had been used to develop the project are using open source. Besides that, the final product for this project is running on a handphone and the solution is by using the author's own handphone which is compatible to run the end product of the project.

CHAPTER 2

LITERATURE REVIEW

2.0   Introduction

The project is about developing a mobile game application that is implemented on handphone. The game is a picture puzzle game which allows the user to challenge his or her mind to unscramble the picture so that the game is completed. The main concern of the developer for developing the application is to understand the Symbian OS which is the operating system for handphone and Java 2 Micro Edition (J2ME) which is the programming language used for developing the application.

2.1   Java 2 Micro Edition (J2ME)

Java 2 Micro Edition is a programming language. As stated by Martin de Jode [1], J2ME has been developed primarily as a technology for the execution of applications on constrained devices. The constrained devices are mobile phones, PDAs, TV set-top boxes, in-vehicle telementary, residential gateways and other embedded devices.

J2ME is the newest and smallest addition to the Java family and it provides a development environment for those small or constrained devices. J2ME is derived from J2SE and it shows all the characteristics of Java language. The figure 2.1 below shows the full J2ME architecture for mobile phones and entry-level PDAs.
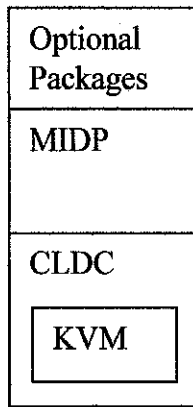
Figure 2.1 The Java landscape

From the J2ME architecture above, it shows that J2ME provides an environment for developers wishing to develop application for small devices which are mobile phones and entry-level PDAs. This environment has been specialized to cater for machines with less capacity. The Figure 2.1 shows that the architecture of J2ME consists of KVM, CLDC, MIDP and Optional Packages.

K-Virtual Machine (KVM) which is used as virtual machine for CLDC. As mentioned by Martin de Jode, KVM is complaint with the Java Virtual Machine Specification and the Java Language Specification. The libraries available are typically split into two categories which are those defined by CLDC and those defined by its optional packages such as MMAPI and WMA.

According to Martin de Jode, Connected Limited Device Configuration (CLDC) is one of configurations in J2ME and it is used for devices with intermittent network connections, small processors and limited memory. Above the CLDC layer, there is one layer that is known as profiles. Profiles provide APIs for a specific class of device that create the ability for each configuration to be adapted and targeted towards vertical markets [1].

6

Here are the profiles that are available in J2ME:

- Mobile Information Device Profile (MIDP)

This profile offers the core functionality required by mobile applications including user interface, network connectivity, local data storage and importantly, application lifecycle management.

- Information Module Profile (IMP)

This profile is based upon the MIDP 1.0 profile. IMP combined with CLDC provides a Java application environment targeted at resource-constrained and embedded network devices. This means that the developers' skills can be easily transferred to IMP.

2.2   Symbian OS

Symbian OS is a mobile Operating System (OS) from Symbian Ltd. According to Leigh Edwards [2], Symbian OS is an open, highly robust Operating System for data-enabled mobile phones. Moreover, Martin de Jode has mentioned that Symbian has become involved in shaping the Micro Edition via the expert groups of the Java Community Process. Symbian recognized the strength of the MIDP movement by including J2ME CLDC/MIDP 1.0 and J2ME CLDC/MIDP 2.0 as its standard Java offering in Symbian OS.

Symbian OS was designed for use in small battery-powered devices with extensive communications capabilities. Its key design features are illustrated below:

- Performance – Designed to maximize battery life through careful device specific power management.
- Multitasking – Telephony, messaging and communications are fundamental components. All applications are designed to work seamlessly in parallel.
- Standards – The use of technologies based on industry standards is a basic principle of Symbian OS, ensuring that applications are interoperable with solutions from other platform vendors.

- Object-oriented software and highly modular architecture.

- Memory management optimized for embedded software environment – very small executable sizes and ROM-based code that executes in place.

- Runtime memory requirements are minimized.

- Security mechanism for enabling secure communications and safe data storage.

- Application supports for an international environment, with built-in Unicode character sets and ease of localization.

2.3   Series 60 Mobile Phone

For the project the handphone that the author will use is Nokia 7610. This handphone is an imaging phone features an integrated 1-megapixel with 4x zoom. The operating system of the Nokia 7610 is Symbian OS v7.0s. Moreover, the Nokia 7610 is Series 60 mobile phone [5].

According the author research, Series 60 Platform builds on the Operating System from Symbian, complementing it with a configurable graphical user interface library and a comprehensive suite application plus other general purposes. According to DIGIA Inc. [3], Series 60 fully supports installing and running applications designed natively for Symbian OS.

2.4   TiledLayer Class

As been mentioned in the chapter 1, the project involves TiledLayer class from the Game API. According to Eric Giguere [4], TiledLayer consists of a grid of cells each of which can be filled with an image tile. This class allows large virtual layers to be created without the need for an extremely large image. The columns and rows arguments represent the number of columns and rows in the grid. Furthermore, the tileWidth and tileHeight arguments represent the width and the height of a single tile in pixels. The image argument represents the image used for creating the set of tiles that will be employed to populate the TiledLayer. The image is broken up into a series of equally-sized tiles. He also has mentioned that by natural, the dimension of

the image in pixels must be an integral multiple of the dimension of an individual tile [4].
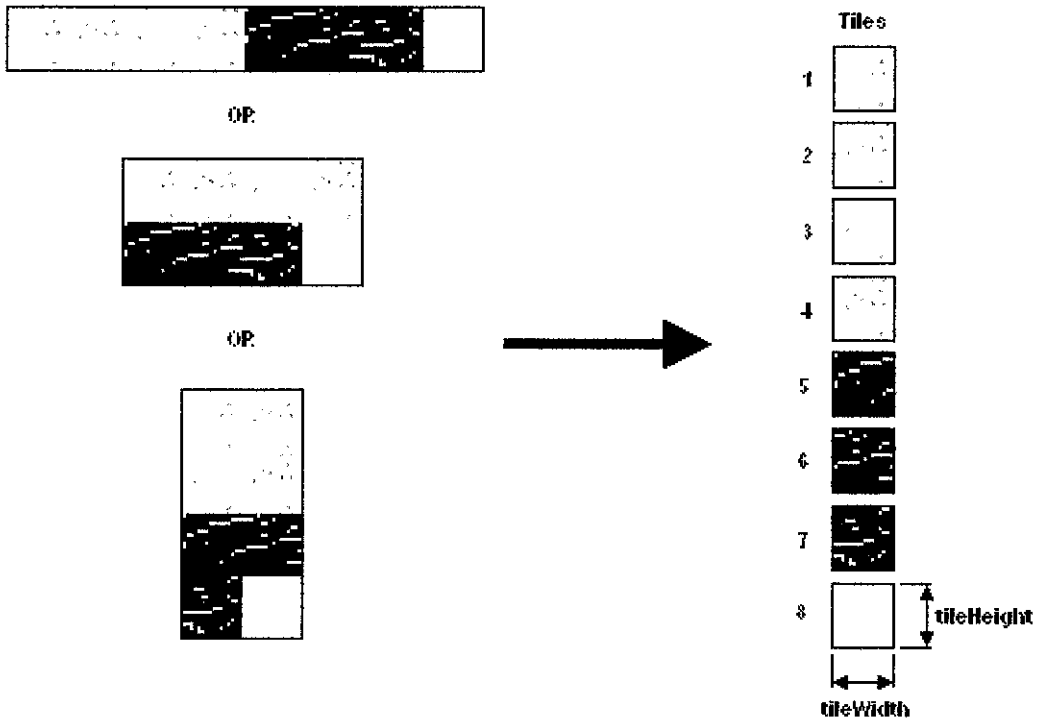
Figure 2.2 Tiles

According to Matteo Zinato [5] and as shown in the figure above, each tile is assigned a unique index number. The tile located in the upper-left corner of the image is assigned an index of 1. The remaining tiles are then numbered repeatedly in row-major order. These tiles are regarded as static tiles because there is a fixed link between the tile and the image data associated with it. Moreover, the same tile set can be stored in several different arrangements depending on what is the most convenient for the game developer.

The next thing is TiledLayer's grid is made up of equally sized cells. The number of rows and columns in the grid are specified in the constructor, and the physical size

of the cells is defined by the size of the tiles. The contents of each cell are specified by means of a tile index. A positive tile index refers to a static tile, and a negative tile index refers to an animated tile. A tile index of 0 indicates that the cell is empty. An empty cell is fully transparent and nothing is drawn in that area by the TiledLayer and by default, all cells contain tile index 0. Several cells may contain the same tile. However, a single cell cannot contain more than one tile.

The following figure shows how a simple background can be created using a TiledLayer.
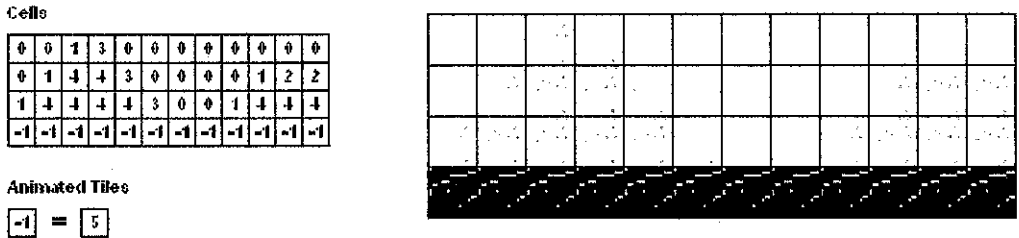
Figure 2.3 Cells

Therefore, as far the project is consent; the TiledLayer class can be used in order to create the grid for the picture so that it can be scrambled.

2.5   MIDP 2.0 and Game API

The project is using MIDP 2.0 as the profile that offers the core functionality required by mobile applications such as user interface, network connectivity, local data storage, and application lifecycle management. One of the most interesting new features is Game API. For this project, the main focus of the study on MIDP 2.0 is Game API. According to Mikko Kontio [6], the Game API helps developers to develop faster user interfaces with better usability. The Game API also helps to reduce the size of the jar file and save device's resources, like memory.

10

Furthermore, based on the article by Mikko Kontio, he has mentioned that the basic idea of Game API is that the game screen consists of layers. Usually the game area is larger than the screen. So, Game API provides a view window. This view window is a view of the whole game area. Besides that, the view window can be moved easily and points in the view window can be referenced as points on the actual screen. The Game API consists of five new classes such as GameCanvas, Layer, LayerManager, Sprite, and TiledLayer. For the project, the TiledLayer is most important class that makes the project success.

CHAPTER 3

METHODOLOGY

3.2 Introduction

This chapter elaborates the methodology used for developing the project. The methodologies used are involved planning, analysis, design and development. Besides that, the system requirements for developing the project and also for implementing the project are discussed further in the chapter.

3.1 Planning

Planning involves developing the plan on what is going to be developed and how the project can be developed and implemented at the final stage.

- Redefine topic – the author had redefined the project scope in order to get clear knowledge on the project that is going to be developed.
- Data gathering – the author had gathered as much knowledge that is related to the topic selected. The knowledge will be analyzed during the analysis stage.
- Outline documentation – the author had outlined the documentation that is going to be created during the project development.
- Decide development tool – the author had decided the tools that are going to be used for developing the project.
- Develop Gantt chart – the author had planned and developed Gantt chart for the project. Refer to Appendix A.

3.2 Analysis

Analysis involves analyzing the knowledge gathered during the planning stage so that only the related information will used for project.

- Analyze development tools – the author had analyzed the tools that will be used whether the tools can accommodate the important in developing a mobile application such as packages from Game API and Mobile Media API.
- Define requirements – the author had defined the project requirements according to standard for a mobile application.

3.3 Design

Design stage is where the project user interfaces and UML class diagram will be designed and created.

- User interface design – the author had designed and determined the user interface of the application.
- Application workflow design – the author had designed the DFD and UML class diagram so that the flow of the application can be understood and this UML class diagram had been used for next stage which is the development stage.

3.4 Development

- Develop user interface – the author had developed the user interface according to designated user interface.
- Develop codes – the author had generated the codes in order to develop the application.
- Testing – the developer has tested the codes developed.
- Implementation – the developer had transferred the game application to handphone and had delivered it as a final product.

3.5   System Requirements

3.5.1   Software

The tools which are required for developing and then implement the project are Nokia PC Suite and J2ME Wireless Toolkit 2.2.

3.5.2   Hardware

The personal computer had been used during the development stage. The technical specs for the personal computer are as below:

Table 3.1 Personal computer technical specs

Operating system	Window XP Home Edition Version 2002 Service Pack 1
Processor	Intel® Pentium® 4 CPU 2.20GHz
Memory	384 Mb of RAM
Disk Space	30GHz

A camera phone is used for implementing the final product. The camera phone that will be used is Nokia 7610. Here are the technical specs of the Nokia 7610:

Table 3.2 Nokia 7610 technical specs

Operating system	Symbian OS v7.0s
Developer platform	Series 60 2nd Edition, Feature Pack 1
Java technology	CLDC 1.0 Wireless Messaging API (JSR-120) Mobile Media API (JSR-135) Bluetooth API (JSR-82 No OBEX) MIDP 2.0 Nokia UI API
Screen display	Color Depth: 16 bit Resolution: 176 x 208
Memory	Unlimited Heap size Shared Memory for Storage: 8 MB Unlimited Jar size
Keypad descriptions	2 labeled soft keys 5-way scrolling Grid key mat
Local and PC connectivity	Bluetooth USB
Extra features	1 Megapixel camera w/4x digital zoom

CHAPTER 4

RESULT AND DISCUSSION

4.0 Introduction

As mentioned in Chapter 3, there are four methodologies involve in the project development. So, this chapter will discuss more on the progress achieved by the author after performing those four stages.

4.1 Data Flow Diagram

Since the author has developed the project, the author has come out with a data flow diagram (DFD) for the project. The DFD has been developed so that the author gets clear vision and understands how the game will be running. A data flow diagram describes a system from the point of view of the data that passed between the processes. It shows the processes (circles or boxes) and the data flowing in and out the processes (represented by named lines). The DFD of the project explains the how the data flow from the player starts the game and until the game is completed.

For the project's DFD, all the data flows, and the data store are described along with the processes that take place in the game. There are five processes involve in the data flow diagram and one data storage. The data flows from the user to the processes in the application. Firstly, the user needs to select whether he or she wants to play the game by using the newly captured image or the saved image. If the user chooses to use newly captured image, the application then collaborates with camera on the handphone in order to take new picture and then, the picture is saved in the RMS storage. Next, if the user chooses to use the saved picture, then the application loads the picture from the database into the application. Now, the user has to select

16

the difficulty level that they want to play. Then, the user also needs to choose whether he or she wants to play with timer or not. If the user selects to play with timer, the user has to play against the time specified. If the user does not want to play with timer, the user is allowed to play the game until the user is able to unscramble the picture. The next process is scrambling the picture according to the selected level and then display it on the screen and the game is now ready to be played by the user. The project's DFD is illustrated below.
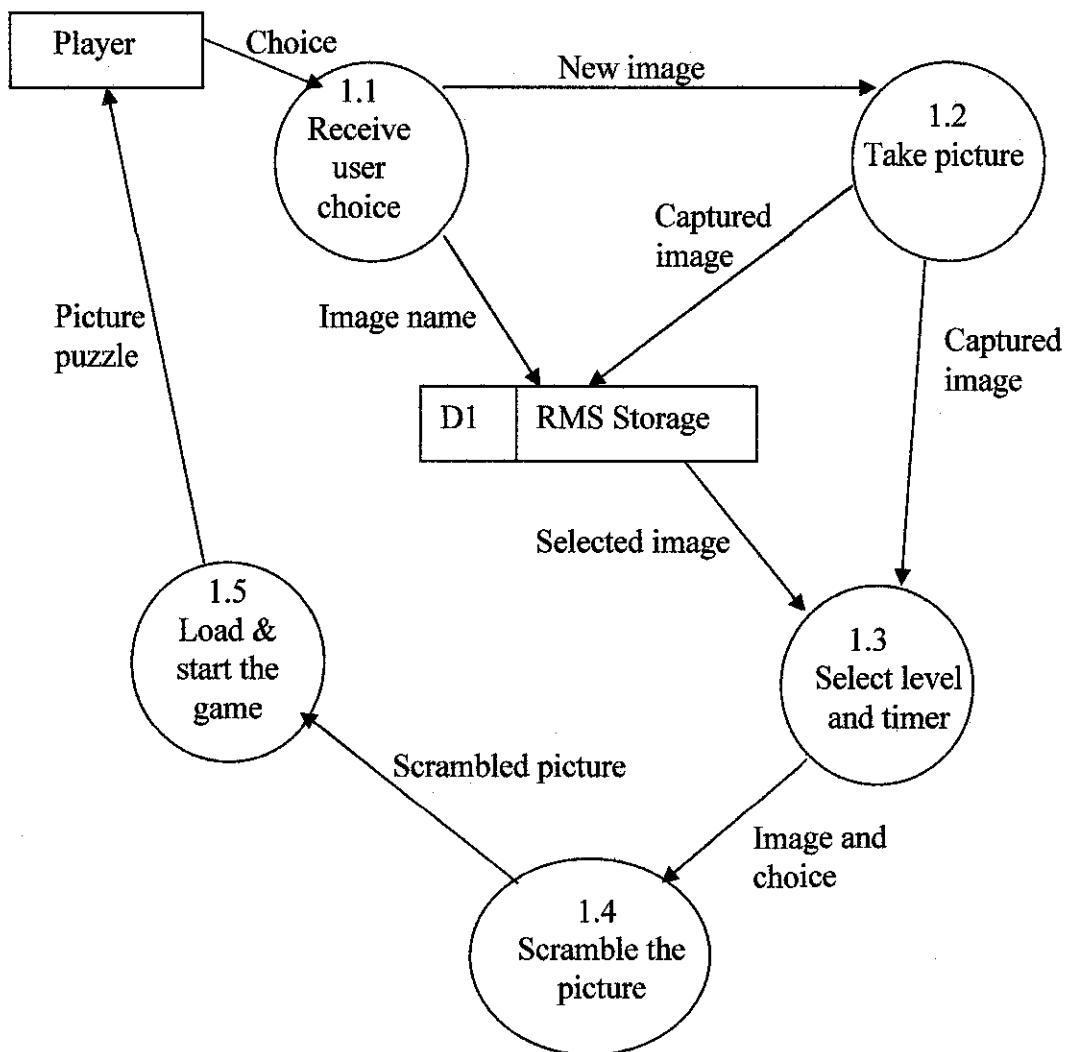

Figure 4.1 Data Flow Diagram (DFD)

4.2   UML Class Diagram

After developing the DFD, the author has created the UML class diagram. A class diagram is a diagram showing a collection of classes and interfaces, and also the collaborations and relationships among the classes and interfaces. Moreover, classes in a class diagram are interconnected in a hierarchical fashion, like a set of parent classes and the related child classes under the parent classes. For the project, the UML class diagram is developed in order to show the author the structure of the game application. So, the author can implement the project efficiently by following the UML class diagram.

Based on the UML class diagram in Appendix B which has been designed by the author, there are eleven author own classes and other three classes are the system packages. The system packages are javax.microedition.lcdui.Form, javax.microedition.lcdui. Textbox, and javax.microedition.lcdui.Canvas. The other eleven classes are:-

- The GameMIDlet class extends MIDlet and it provides implementations for the MIDlet lifecycle methods. The MIDlet life cycle consists of pause, start and terminates the application. This class also acts as application controller since it provides a number of callback methods that are invoked by the application's user interface objects in response to user interaction.

- The ChoiceForm class loads the names of any stored images and displays them in an instance of ChoiceForm.

- The Capturer class is for creating and initializing the VideoPlayer. The VideoPlayer is used to indicate that the data source is the phone's camera and then get the VideoControl and initialize with the CaptureCanvas class. The picture is taken and returns it as a PNG image.

- The CaptureCanvas class provides the Canvas onto which the output of the camera is made. When the user satisfied with the scene, the picture is taken by selecting the "Capture" command.

- The ImageNameBox class extends TextBox and provides an area for the user to enter a name for the new image. In this class when the user selects the "Save" command, the image data and the name are saved to the RMS store.

18

- The PuzzleCanvas class is where the board of the game is created. The user's selected level is used in a method called createBoard() in the class in order to create the scrambled puzzle. This method implements the TiledLayer algorithm and also the timer function.

- The PuzzleCanvasXTime is a class that does the same work just like PuzzleCanvas class but the only different is in this class the timer function is not being implemented.

- The RMSHandler class which handles loading previous image from storage, saving a new image to storage and deleting images from storage. The RMS record store provides persistent storage of data in the form of records within the RecordStore. In this project, there are two RecordStore, one for storing the image data and an index RecordStore. The index RecordStore is for storing the image's name and the record ID.

- The Level class and SelectTime class are used for handling the user's selections which are difficulty levels and timer options.

- The GameHelp class is created for providing the user with the game information and game instructions.

4.3 System Functionalities

As mentioned earlier, the author wants to upgrade the current puzzle game. As a result, the author has able to come out with the newly enhanced picture puzzle game. The game has a few functionalities. The functionalities of the game application are make used the camera build in the handphone to capture picture for the game, save the pictures taken, delete the pictures saved in the RMS storage, three levels of difficulty for the game, options for game timer and hint function. The screenshots of the game can be seen in Appendix C.

- Make use of camera build in the handphone

This picture puzzle game application uses the picture taken by camera on the handphone to create the scrambled picture. Since that, the game application is collaborated with the camera build in the handphone. When the user starts the game, he or she has to choose whether he or she want to capture new image or use the

19

stored images. If the user chooses to capture new image, the application works together with the phone's camera in order to take picture.

- Save and delete picture

The save and delete functions is for the user to organize the pictures stored in RMS storage. The new image is saved into the RMS storage and all images stored in the storage can be used again when the user selects the image's name as he or she play the game again.

- Three levels of difficulty

The levels for the game consist of beginner, intermediate and expert. The game allows the user to choose which level he or she want to play. The levels are used to determine the difficulties that the user has to solve so that he or she can complete the game. This means that the levels are used to determine the grid size that will be applied for scrambling the picture such as beginner level is using grid size of 4x4, intermediate level is using grid size of 5x5 and expert level is using grid size of 6x5.

- Timer function

The final function is options for game timer which allow the user to choose whether they want to play the game with timer or not. When the user chooses to play with timer, the user has to play against time. The timer is determined according to the difficulty level that the user has chosen earlier. For beginner level, the timer is set to four minutes, the timer for intermediate level is 3 minutes and the timer for the expert level is two and half minutes. Furthermore, if the user chooses not to play with timer, the timer is off and the user can play without a time limit until he or she finish the game.

- Hint function

This game application also provides hint function for the user to view again the original image for 5 seconds in order to help the user solved the puzzle.

4.4   TiledLayer Class

TiledLayer class is a predefined class from Game API. As mentioned in the literature review, the TiledLayer class consists of a grid of cells and these cells can be filled with an image tile. So, the TiledLayer class had been used for the project in order to make the picture puzzle. In the project, the TiledLayer class brakes the picture taken into a series of equally-sized tiles. The width and height of a single tile is in pixel and for the project, the width and height of a tile is determined according to the level of difficulty that the user has chosen. The actual size of the picture taken for the game is 160x120 pixels. Therefore, the width and height of a single tile in beginner level is 40x30 pixels, intermediate level is 32x24 pixels and expert level 32x20 pixels. All tiles are assigned with a unique index number.

Next thing is TiledLayer's grid. The TiledLayer's grid is made up of equally size cells which consist of rows and columns. Since the project has three types of difficulty level, the TiledLayer's grid is divided into a few sizes. The size for beginner level is 4x4 grids, intermediate level is 5x5 grids, and expert level is 6x5 grids. Moreover, the physical size of the cells is defined by the size of the tiles and the contents of each cell are specified by the means of tile index number.

Now, the grid of cells is filled with the image tiles. For scrambling the picture, the author had used another predefined class in Java which is Random class. The Random class is used to jumble up the tiles within the grid. While the user plays the game, the application will determine whether the current arrangement of tiles is equal to the original image and hence the game has been successfully completed.

4.5   Limitations

- The game application does not have a ranking function so that it can store the achievement of the user of the game.

- The game application is for single player only.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

In conclusion, the developed game application is able to achieve its' objectives and it is also able to run as expected when the game application is being implemented on the handphone. The Picture Puzzle is expected to be one of the game options in handphone that has camera feature. Picture taken using camera phone is no longer for printing, sending to others and transferring to personal computer but the picture is used as a medium for entertainment and enjoyment to the handphone user.

5.2 Recommendations

Based on the project limitations, there are some recommendations and improvements that can be done to the project in the future. Here are the recommendations:

- Ranking function can be implemented in the game since the developed game application does not have ranking function. The ranking function should allow the user of the game to save his or her achievement in terms of the user's name and the time taken to complete the game.
- The game application can be enhanced to more interactive game. In the future, the game shall be able to allow another person who has the same game application to take picture using his or her handphone and the picture taken can be sent to his or her colleague who also has the same game application. Then, the colleague can play the game by using the sent picture.

REFERENCES

[1]   Jode, M. Programming Java 2 Micro Edition on Symbian OS, John Wiley & Sons, Ltd, 2004

[2]   Edwards, L., Barker, R. and the Staff of EMCC Software Ltd. Developing Series 60 Application, Addison-Wesley, 2004

[3]   DIGIA Inc. Programming for Series 60 Platform and Symbian OS, John Wiley & Sons, Ltd., 2004

[4]   Giguere, J. Java 2 Micro Edition, John Wiley & Sons, Ltd., 2000

[5]   Zinato, M. "J2ME MIDP 2.0: Background and Tracks with TiledLayer." http://www.j2me-guru.com/guru/midp20/j2me_guide_midp2_10.asp, 2002-2005

[6]   Mikko Kantio, "MIDP 2.0: The Game API." http://www.microjava.com/articles/techtalk/game_api , 2003

[7]   Forum Nokia, "Device Details---Nokia 7610." www.forum.nokia.com, 2004

[8]   Sun Microsystems, Inc. "Java Verified Program for J2ME." www.javaverified.com, 1995 - 2004

[9]   Jode, M. "Symbian on Java" www.symbian.com/technology/standard-java.html, 2004

[10]   Symbian, "Symbian Phones" www.symbian.com/phones, 2006

[11]   Feng, Y. and Zhu, J. Wireless Java Programming with Java 2 Micro Edition, Sams Publishing, 2001

[12]   Wu, C.T. An Introduction to Object-Oriented Programming with Java, Mc Graw Hill, 2004

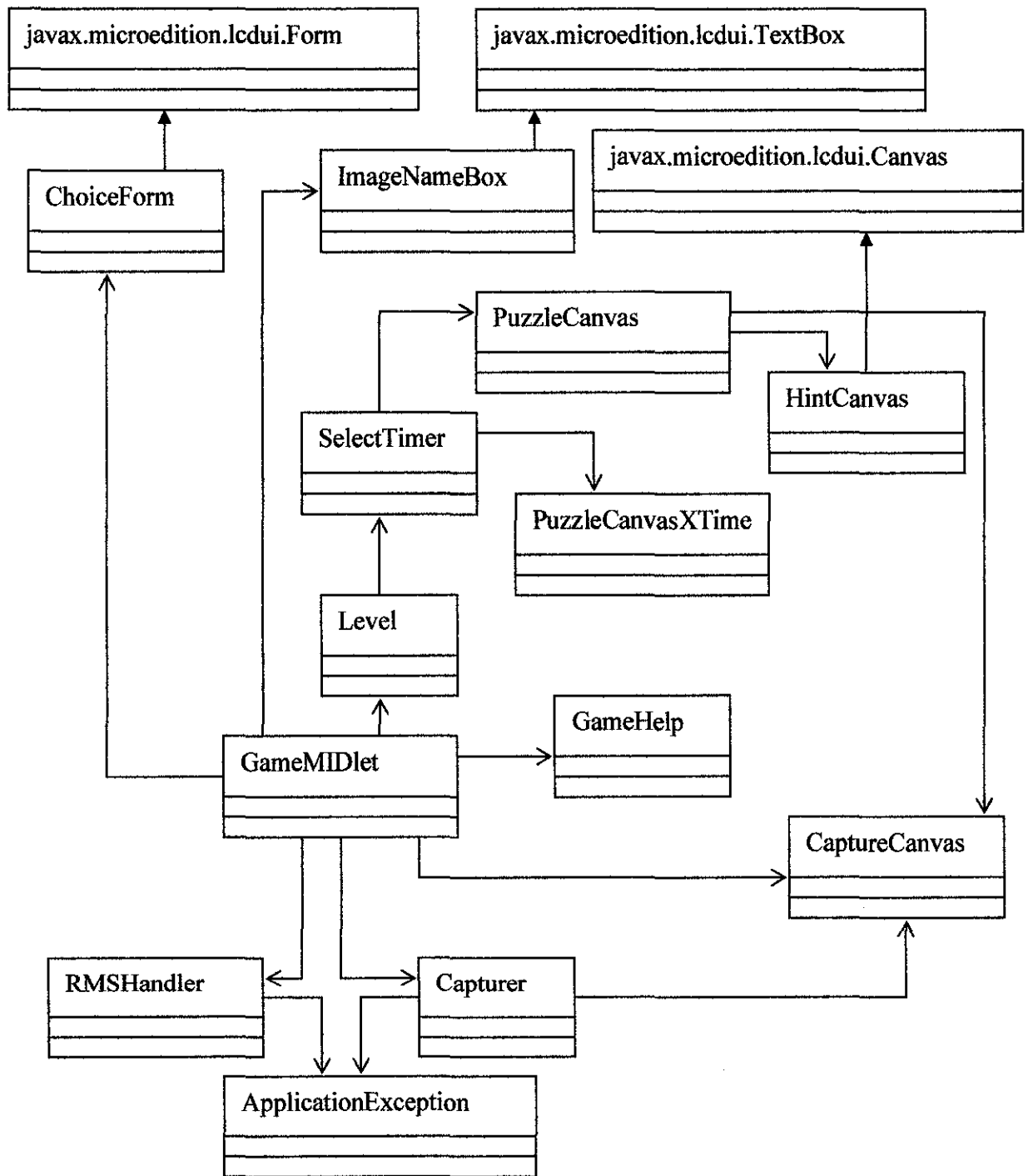[13]   Wells, M. J. J2ME Game Programming (Game Development), Thomas Course Technology, 2004

APPENDICES

APPENDIX A: Gantt chart

ID	❸	Task Name	Duration	Start	Finish	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	Jun	Jul
1	✓	Proposal submission and approved	21 days?	Mon 8/22/05	Mon 9/19/05												
2	✓	**Planning phase**	**9 days?**	**Tue 9/20/05**	**Fri 9/30/05**												
3	✓	Redefine topic	1 day?	Tue 9/20/05	Tue 9/20/05												
4	✓	Plan on the tools to be used	9 days	Tue 9/20/05	Fri 9/30/05												
5	✓	Plan on research and study process	9 days	Tue 9/20/05	Fri 9/30/05												
6	✓	**Analysis and Design phase**	**16 days?**	**Fri 9/30/05**	**Fri 10/21/05**												
7	✓	Research and study process	15 days	Fri 9/30/05	Thu 10/20/05												
8	✓	System requirement	4 days	Fri 9/30/05	Wed 10/5/05												
9	✓	DFD and UML class diagram design	5 days	Thu 10/6/05	Wed 10/12/05												
10	✓	Interface design	4 days	Tue 10/18/05	Fri 10/21/05												
11	✓	Preliminary Report submission	1 day?	Fri 10/21/05	Fri 10/21/05												
12		**Development phase**	**147 days?**	**Fri 11/11/05**	**Mon 6/5/06**												
13	✓	Interim Report submission	1 day	Fri 11/11/05	Fri 11/11/05												
14	✓	Developing codes	90 days	Fri 11/11/05	Thu 3/16/06												
15	✓	Implement the application	1 day?	Fri 3/17/06	Fri 3/17/06												
16		Testing	30 days	Fri 3/17/06	Thu 4/27/06												
17	✓	Oral presentation part 1	2 days	Mon 12/5/05	Tue 12/6/05												
18	✓	Progress Report submission	1 day?	Mon 2/13/06	Mon 2/13/06												
19	✓	Pre-EDX	1 day?	Fri 3/3/06	Fri 3/3/06												
20	✓	First Draft submission	1 day	Fri 5/5/06	Fri 5/5/06												
21	✓	Final Report submission	1 day?	Fri 5/19/06	Fri 5/19/06												
22	▥	Final Dessertation submission	1 day?	Mon 6/5/06	Mon 6/5/06												
23	▥	Oral presentation part 2	1 day?	Mon 6/5/06	Mon 6/5/06												

Project: dateline
Date: Sat 6/17/06

Task		Milestone ◆	External Tasks
Split	....	Summary ◥◤	External Milestone ◈
Progress	▬▬▬▬	Project Summary	Deadline ⇩

Page 1

APPENDIX B: Project UML Class Diagram

```
┌─────────────────────────────────┐        ┌─────────────────────────────────┐
│ javax.microedition.lcdui.Form   │        │ javax.microedition.lcdui.TextBox│
├─────────────────────────────────┤        ├─────────────────────────────────┤
│                                 │        │                                 │
├─────────────────────────────────┤        ├─────────────────────────────────┤
└─────────────────────────────────┘        └─────────────────────────────────┘
            ▲                                           ▲
            │                                           │
            │                    ┌──────────────────┐   │   ┌───────────────────────────────────┐
┌───────────────────┐           │  ImageNameBox    │───┘   │ javax.microedition.lcdui.Canvas   │
│   ChoiceForm      │───────────▶├──────────────────┤       ├───────────────────────────────────┤
├───────────────────┤           │                  │       │                                   │
│                   │           ├──────────────────┤       ├───────────────────────────────────┤
├───────────────────┤           └──────────────────┘       └───────────────────────────────────┘
└───────────────────┘                                                       ▲
     ▲                               ┌──────────────────┐                   │
     │                               │  PuzzleCanvas    │───────────────────┤
     │                          ┌───▶├──────────────────┤                   │
     │                          │    │                  │          ┌──────────────────┐
     │                          │    ├──────────────────┤          │   HintCanvas     │
     │                          │    └──────────────────┘          ├──────────────────┤
     │                  ┌──────────────────┐                       │                  │
     │                  │   SelectTimer    │──────────┐            ├──────────────────┤
     │                  ├──────────────────┤          │            └──────────────────┘
     │                  │                  │          ▼
     │                  ├──────────────────┤    ┌──────────────────┐
     │                  └──────────────────┘    │ PuzzleCanvasXTime│
     │                          ▲               ├──────────────────┤
     │                          │               │                  │
     │                  ┌──────────────┐        ├──────────────────┤
     │                  │    Level     │        └──────────────────┘
     │                  ├──────────────┤
     │                  │              │
     │                  ├──────────────┤        ┌──────────────────┐
     │                  └──────────────┘        │    GameHelp      │
     │                          ▲               ├──────────────────┤
     │                          │               │                  │
┌──────────────────┐            │               ├──────────────────┤
│   GameMIDlet     │────────────────────────────▶└──────────────────┘
├──────────────────┤                                           ┌──────────────────┐
│                  │                                           │  CaptureCanvas   │
├──────────────────┤───────────────────────────────────────────▶├──────────────────┤
└──────────────────┘                                           │                  │
   │      │      │                                             ├──────────────────┤
   │      │      │                                             └──────────────────┘
   ▼      ▼      ▼                                                      ▲
┌──────────────┐   ┌──────────────┐                                    │
│ RMSHandler   │◀─▶│   Capturer   │────────────────────────────────────┘
├──────────────┤   ├──────────────┤
│              │   │              │
├──────────────┤   ├──────────────┤
└──────────────┘   └──────────────┘
        │      │
        ▼      ▼
   ┌──────────────────────┐
   │ ApplicationException │
   ├──────────────────────┤
   │                      │
   ├──────────────────────┤
   └──────────────────────┘
```
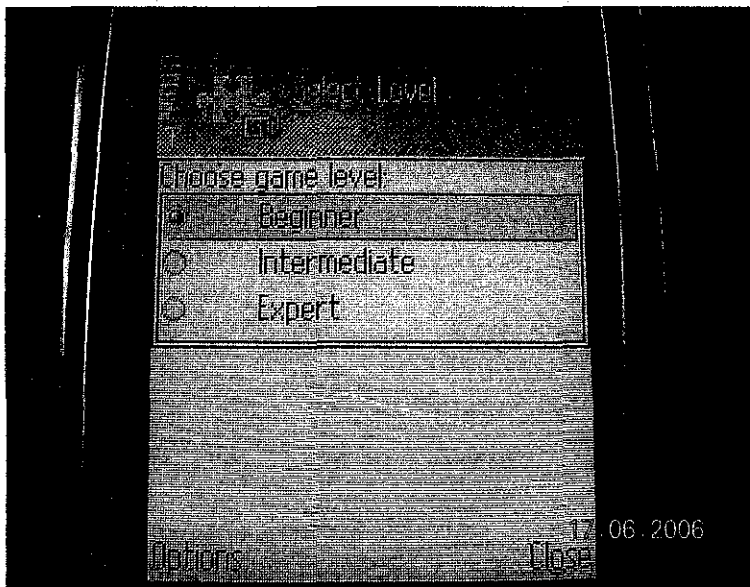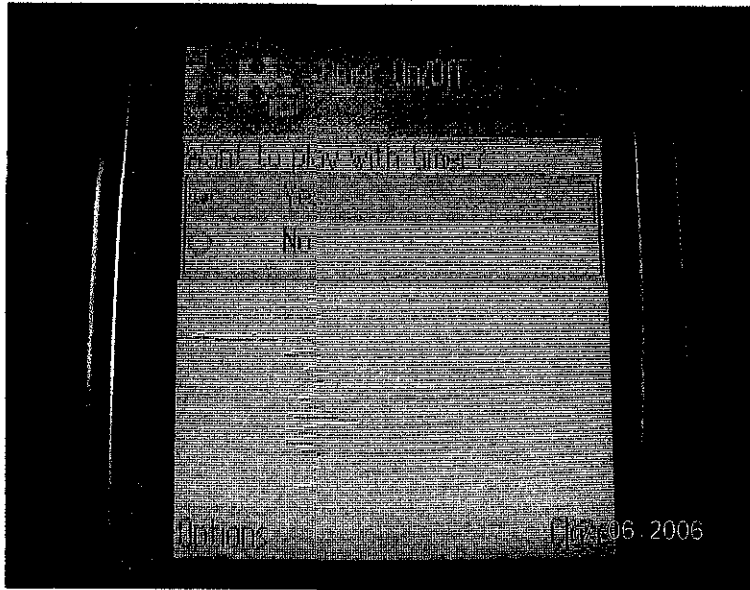
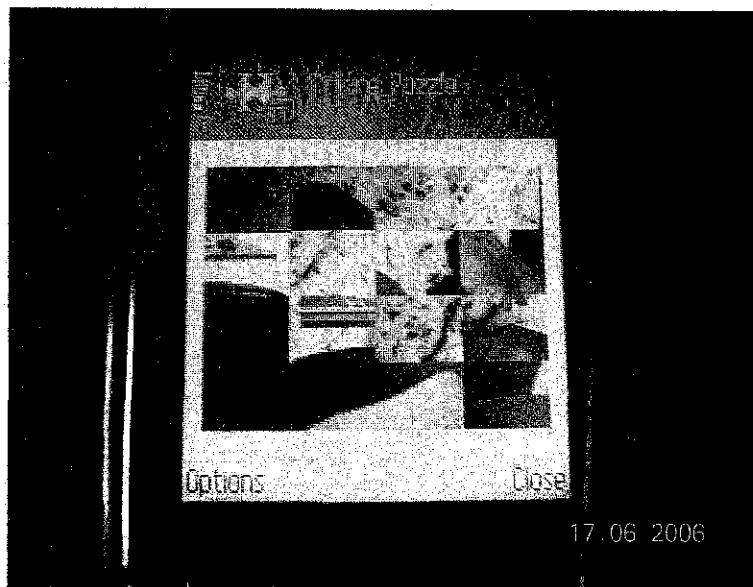# APPENDIX C: Screenshots
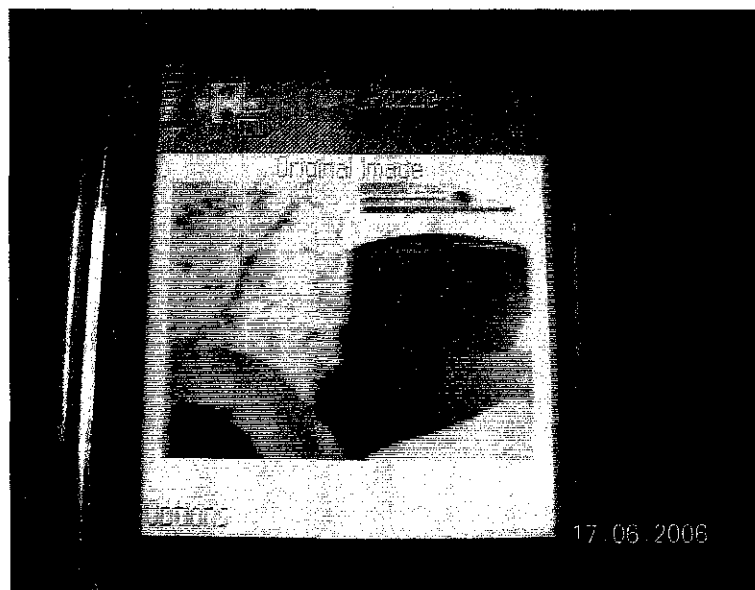
1. Choose image option
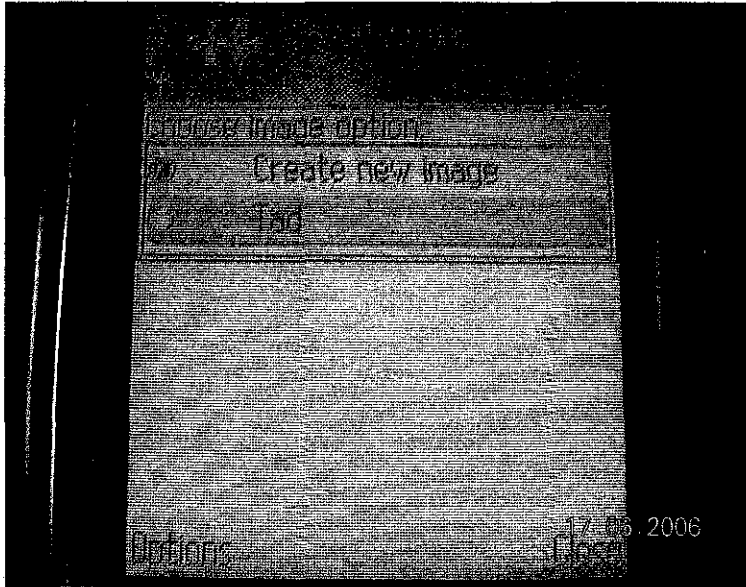
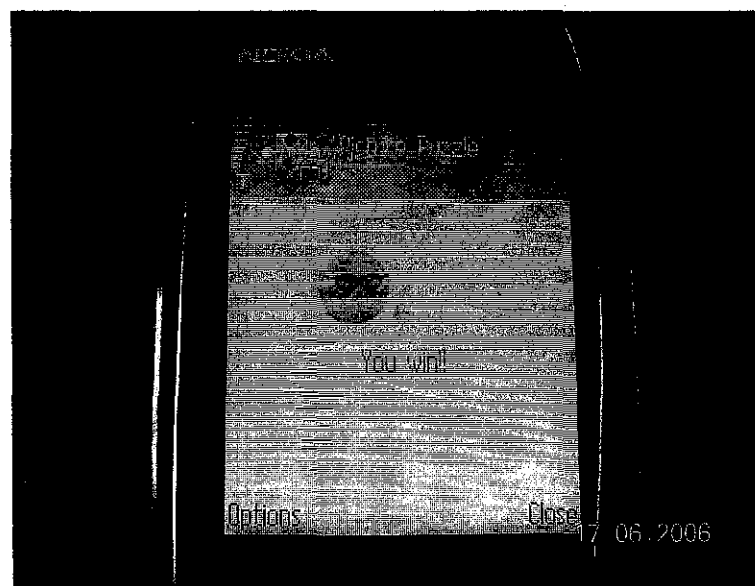# 2. Choose game level
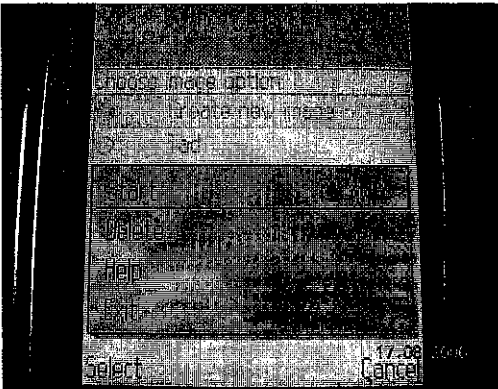
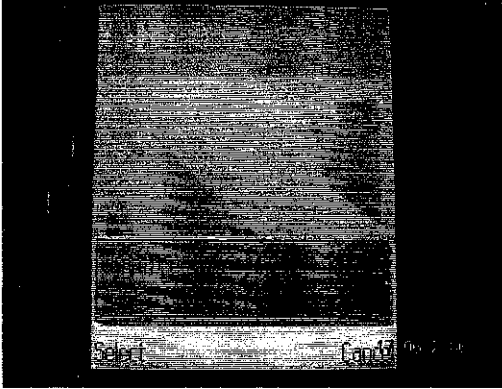## 3. Play with / without timer
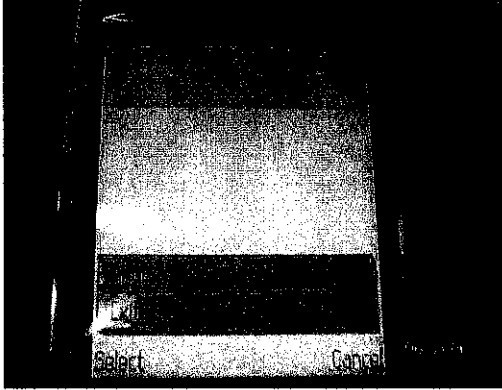

4. Solve the puzzle
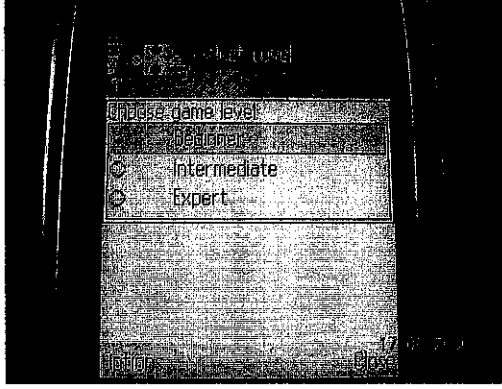
5. Hint
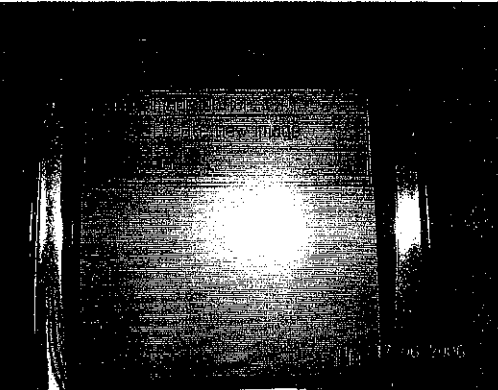
6. Solved the puzzle

APPENDIX D: User Manual

User Manual

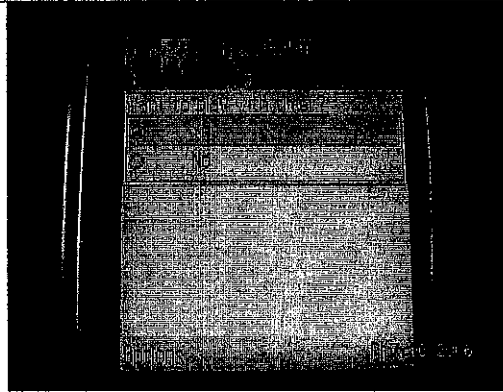| Step	Description
1. Choose image option	Choose to create new image or to use available image. Scroll the joystick on keypad to choose the image option and press the center button of the joystick to select the option. To use new image for the game, choose "Create new image" and to use available image, choose the image name display on the screen. Then press "Options". Choose "Start" and then press "Select".

2. Capture image	
	Focus at any object and then press "Options".
	Next, press "Capture".
	Remember: Do not move the handphone while taking the picture.
3. Save the picture taken	
	Insert any text as the image name and then press "Options".
	Next, choose "Save" and press "Select".
4. Choose game level	
	Choose game level ("Beginner", "Intermediate", "Expert").
	Scroll the joystick on keypad to choose the game level and press the center button of the joystick to select the level.
	Press "Options".
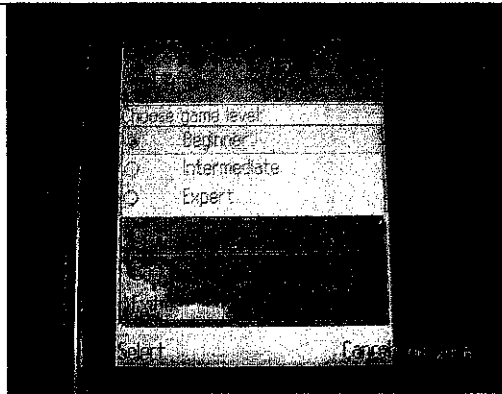
|	
	Choose "Start" and press "Select".
5. Choose timer	Choose to play with timer or not ("Yes" or "No"). Scroll the joystick on keypad to choose play with timer or not and press the center button of the joystick to select the selection. Press "Options". Choose "Start" and press "Select".
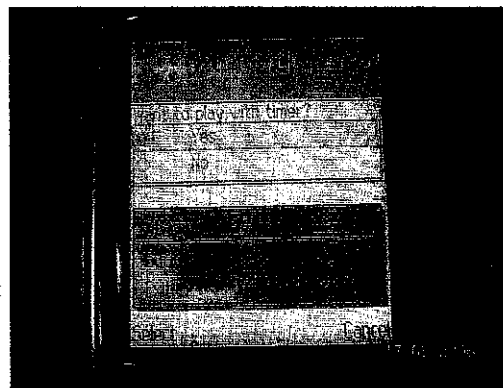
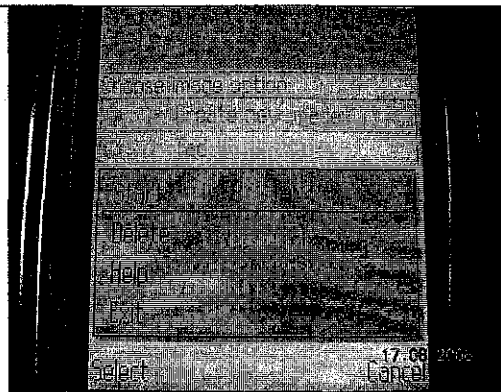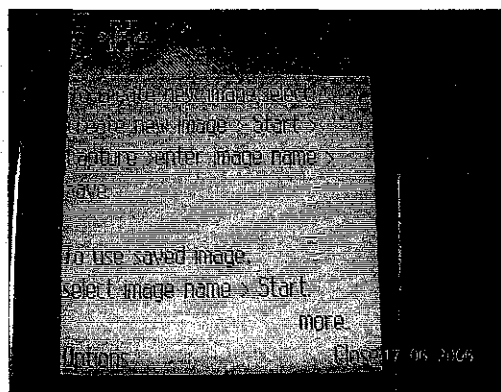| 6. | Solve the puzzle |
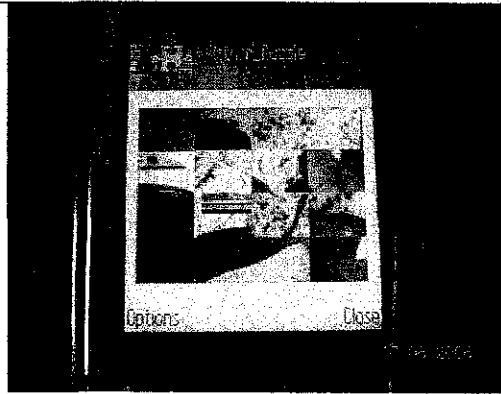
To move the red cursor:-

Move left – press key 4

Move right – press key 6

Move up – press key 2

Move down – press key 8

Select the tile – press key 5

To get hint, press "Options".
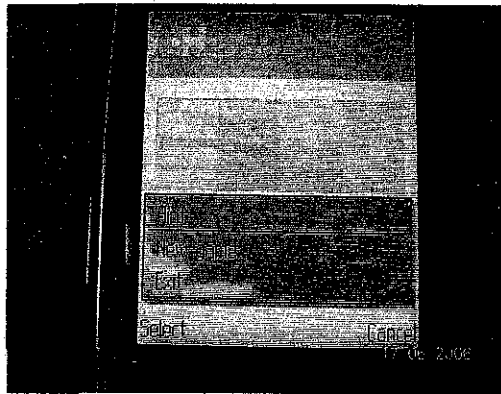
Choose "Hint" and then press "Select". | |

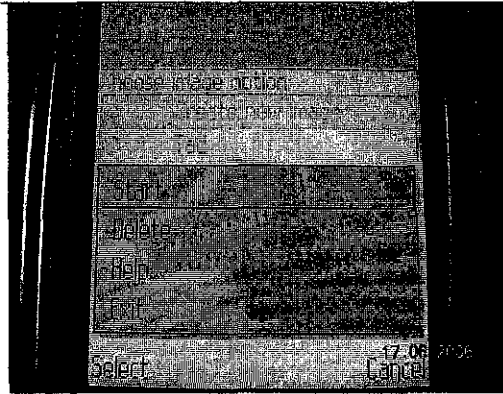|		
7. Play new game		
	Press "Options". Choose "New game" and press "Select".	
8. Help		
	Press "Options". Choose "Help" and then press "Select". 	

| 9. Delete image | |
| | Select image name to be deleted and press "Options".

Choose "Delete" and then press "Select". |