SUDOKU HELPER

By:

Muhammad Asyraf bin Abdul Razak
(16252)

January 2015

Dissertation submitted in partial fulfilment of
the requirements for the
Bachelor of Technology (Hons)
(Information and Communication Technology)

Universiti Teknologi Petronas,
31260, Bandar Seri Iskandar,
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

SUDOKU HELPER

by

Muhammad Asyraf Bin Abdul Razak
16252

Dissertation submitted in partial fulfilment of
the requirement for the
Bachelor of Technology (Hons)
Information and Communication Technology

Approved by,

_____

(Saipunidzam Mahamad)

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK
Jan 2015

CERTIFICATION OF ORIGINALITY

This is to verify that author is responsible for the work submitted in this project, that the original work is written by author except as specified in the references. It is also to ensure that this work content have not been undertaken or done by unspecified sources or persons.

(MUHAMMAD ASYRAF BIN ABDUL RAZAK)

ACKNOWLEDGEMENT

First of all, author would like to thank Universiti Teknologi Petronas for giving the opportunity to do a final year project. This project is important for author to implement all of the knowledge that author had learnt for the past 4 years. As a student, it is also a new opportunity to learn a new thing in order to become a well-rounded undergraduate. Besides that, this final year project also gave more value to the potential employer in the future.

In addition, author would like to thank Computer and Information Science department (CIS) for their cooperation to guide author to complete this project. Author also would like to express sincere gratitude to his final year project supervisor, Mr. Saipunidzam Mahamad who had spent a lot of his time and effort by giving some advices, ideas and morale support.

Author also want to thank his family for all their supports and advices during the process of completing this project. Not to forget all of author friends who help him in so many ways. At last, author want to thanks for those people who had been involves in the making of this final year project such as coordinator and lecturers

ABSTRACT

In this paper research, author presents an algorithm to provide a tutorial for any Sudoku player who got stuck during the solving process. With this ‘Sudoku Helper’ application, author wants to encourage the Sudoku player to complete the puzzle without looking to the full answer scheme. Thus, it will be more challenging and fun for the user. The project is done by author using a Java programming language in Android Studio.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Background Study

Sudoku is one of the most popular games in the world nowadays and had been played by millions of people. This is because of its simple rules and complex problem-solving which give challenges and satisfaction for player that can complete the puzzle. Sudoku was introduced by Nikoli, Japanese puzzle company in 1986 [1]. It is a logic puzzle game that involves numbers and normally play with 9 x 9 grid. The sole goal for Sudoku puzzle is to write down a number from 1 to 9 in such a way that each number must appear once in every row, column and 3x3 block. The puzzle starts with a given clues or digit in various position as stated in Figure 1.1.

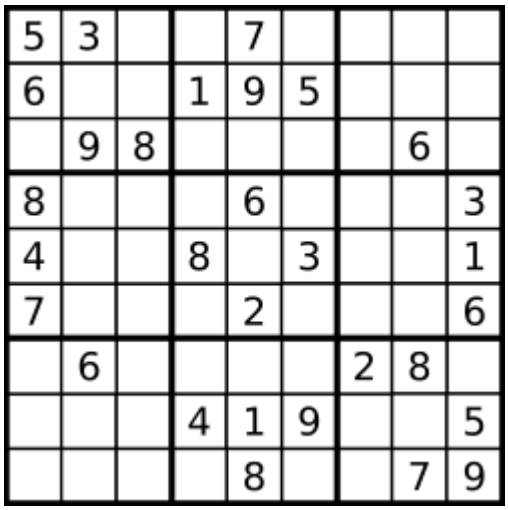


Figure 1.1. A common Sudoku puzzle

The job of Sudoku player is to fill up the empty space with numbers until the puzzle is complete. There are a lot of algorithm that can be used to start fill up the empty spot until the puzzle is completed. This algorithm will be discussed by author in the later section. A Sudoku puzzle must be start with a minimum number of 17 clues. That means a computer cannot generate a complete solution if the puzzle contain 16 or less clues [2]. The example of Sudoku above contains 30 clues and can be classified as 'easy' level. The number of clues will be decreasing as the level of hardness increasing.

Figure 1.2 below is the complete answer scheme for Sudoku puzzle in Figure 1.1. It is to give an overview for beginner player to see how this puzzle is done.

5	3	4	6	7	8	9	1	2
6	7	2	1	9	5	3	4	8
1	9	8	3	4	2	5	6	7
8	5	9	7	6	1	4	2	3
4	2	6	8	5	3	7	9	1
7	1	3	9	2	4	8	5	6
9	6	1	5	3	7	2	8	4
2	8	7	4	1	9	6	3	5
3	4	5	2	8	6	1	7	9

Figure 1.2. Completed Sudoku

1.2 Problem Statement

There are numerous Sudoku solver applications that can be found online whether it is free or paid. However, most of the application designed is to find a complete solution with the fastest time. There are no applications that can provide a step by step tutorial process of completing Sudoku puzzle for beginner player.

1.3 Objective

Human brain consists of millions of nerve cell or basic information processing unit called neurons. By using multiples neuron simultaneously, the brain can perform its function much faster. Strong connection between neurons leading to the ‘right answer’ while those leading to ‘wrong answer’ are weak connection. Thus, author develops this application and targeted beginner player as its main user so that it can help their brain to work faster in early stage.

The main objective for the author to develop this application is :-

i. To develop an application that can provide hint or act as a helper for Sudoku player.

ii. To provide mechanism and platform for beginner player to learn Sudoku using backtracking algorithm.

1.4 Project Scope

Project scope defines what is needed in order to create and complete a certain project. In this particular project, the project mainly focused on the searching algorithm that can be used to detect the number of clues on the beginning, generate solution when needed and prompt out the guide to the player. There are several algorithms that can be used in this particular topic such as hidden single strategy, packet, hitting algorithm, differential algorithm, brute force algorithm, backtracking algorithm and constraint propagation.

1.5 Relevance and Feasibility

This application is designed to help primary school student to play Sudoku as it can help to train their brain in the young age. Sudoku does not require complicated technique or mathematical algorithm but the player just need to find differences in each row, column and block. In order to solve Sudoku puzzle, human brain uses a set of neural pathways called associative memory which help the player to identify pattern from the given clues in Sudoku puzzle. While playing Sudoku, human memory and logic work together as the memory used to memorize the number and its position and on the other hand, logic help the player to figure out the right answer for the blank space.

In addition, playing Sudoku in the book is quite boring and player tends to feel sleepy. With this application, any Sudoku player can play from their smartphone or iPad which is quite convenience and easier to carry along.

CHAPTER 2

LITERATURE REVIEW

2.1 Algorithms

There are many ways that human brain can solve a Sudoku puzzle. Author will discuss about that in this chapter and show some examples that how a puzzle can be solved. These are a few algorithms that had been discussed by author and there are more out there. Every algorithm had different approach, technique and processing speed. As the technology advanced, some of the algorithm can even provide the Sudoku solution in millisecond.

Hidden Single Strategy

Author would like to discuss about the first algorithm that can be used to solve Sudoku called 'hidden single strategy'. Let first take a look at the Sudoku Puzzle A below.

7	6		8	1			4	5
		5				6	8	
	1	2				3	9	
					4			
1				7				3
		3	5					
	3					4	1	
	9	7				5		
6				3				2

Figure 2.1. Sudoku Puzzle A

First of all, Sudoku player must find any row, column or block that has high amount of number. In this example, row have a lot of number except 2,3 and 9. Since column 3 and 7 contain number 3, thus 3 must be put in sixth cell in the first row. This is how hidden single strategy algorithm can help player to fill in the blank space in the puzzle

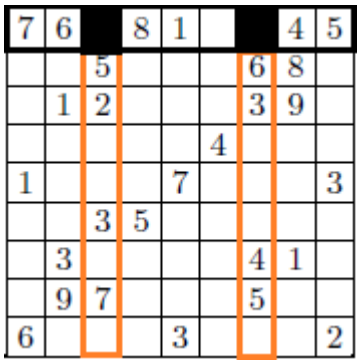


Figure 2.2. Hidden Single Strategy

Packet

People normally used pencil and eraser while playing with Sudoku puzzle. People often look at the cell and make note which number cannot be put into that cell. A *packet* act as reminder to the player and be representation that a number cannot be placed in a cell [3]. For example, if a player determine that number 2,5,7 and 9 cannot be in the first cell in the first row, he/she need to put 4 symbol in the first cell as shown in Figure 5.

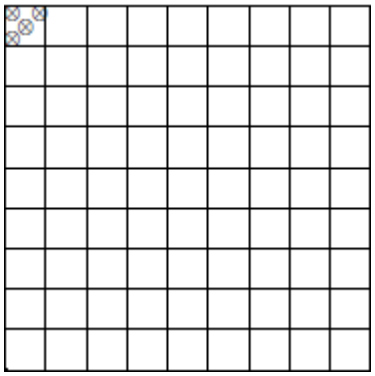


Figure 2.3. Packet in the first cell

Hitting Set Algorithm

This method can be used by using a software that was released in 2006 called ‘checker’. It is a computer-assist algorithm that can be used to calculate mathematical theorem. *Checker* will perform an exhaustive search based on 9x9 grid Sudoku puzzle and came

out with the solution to that puzzle. It is a principle that applicable to any instance of hitting set problem, whether it is decision version (determine if the hitting of a given size exist) or optimization version (finding a smallest hitting set) [4].

Differential Evolution

Differential Evolution (DE) is a technique developed to iteratively improve a candidate solution by given a certain clues such as Sudoku puzzle. This technique makes a few assumptions or no about the clues being optimized and can search a very large space to complete the puzzle. The pseudocode of Differential Algorithm is as follows :-

1. Create the initial population of the genotype.
2. Set the generation number to zero. g=0.
3. Run the software until the stop criteria is met [5].

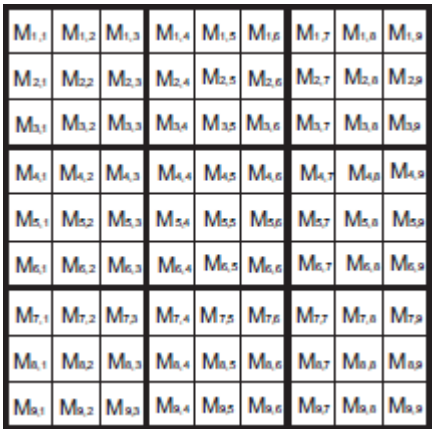


Figure 2.4. Transform the puzzle into genotype

Brute Force Algorithm

This algorithm basically will try to find every possible scenario in a problem until it finds the solution. This technique starts by fill in the 1 and locked the numbers that was part of the puzzle. It then will start changing the first number that is not locked [6]. For every increment, the number will be increase from 1 to 2 and so on until the puzzle is

completed. If it not, the number will be reset to 1 and the second set of increment will be run until it get the solution to the Sudoku puzzle.

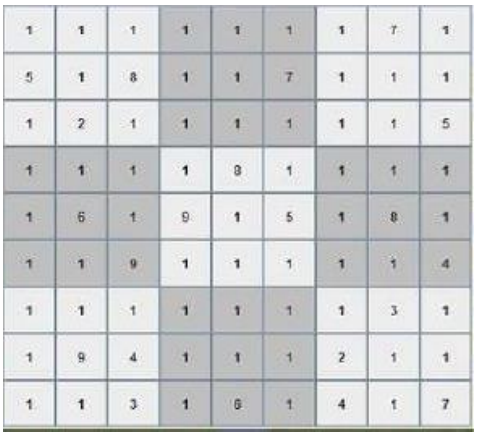


Figure 2.5. First step of Brute Force algorithm

Backtracking Algorithm

This technique is quite similar to Brute Force algorithm but the differences is Backtracking algorithm incrementally build clues for the solutions and abandon each partial clues as soon as it determine that candidate clues is not the right solution. It will try with different number at first, if it fails, then it will backtrack and tries with another different numbers. However, this technique is much faster compared to Brute Force algorithm. It takes more than 90 minutes to find a Sudoku solution using Brute Force algorithm for three different level of difficulty which are easy, medium and hard. On the other hand, it takes 6miliseconds, 42miliseconds and 46miliseconds to solve each easy, medium and hard puzzle by using Backtracking algorithm [6]

. Constraint Propagation

Constraint Propagation involve 2 rules in order to complete a Sudoku puzzle. The rules are as follows :-

1. If a number already exist in the cell for any row, column or block (the cell peers), the possibility of that number for that cell is removed.

2. If all of its peers have a specific number removed, the cell itself must contain that number [7].

It can be explain with the example below. Since number 3 already exist in row 2&3 and column 2&3, then the only place for number 3 is in the first cell of row 1. The cells that fall within the blue line is called cell peer.

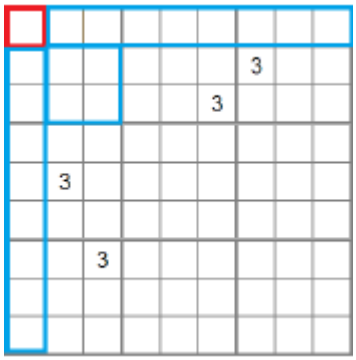


Figure 2.6. Constraint Propagation algorithm

2.2 Comparative Study

The comparative study is a study in which participant assigned different treatment for purposes of comparing the materials [8]. It is to analyze every research papers that had been chosen and the current Sudoku mobile application that already been developed. There must be some research or application that had been designed in the past and author will do a comparison between those applications.

Research Comparative Study

Algorithm	***Processing Speed***	***Limitation***	***Design***
Human Solving Symmetry	Based on human skills	It can solve the puzzle but it takes quite some times.	Hidden Single Strategy approach Packet approach
Hitting Set Algorithm	A few hours	Need to use 'checker' software	Run through the catalogue and search every grid in turn using *checker*.
Differential Evolution	15000 iterations for 'hard'Sudoku	-	Used a genotype strategy (mutation)
Brute Force Algorithm	More than 90 minutes	-	Use node as state in the search algorithm
Backtracking Algorithm	46 milisecond	-	Use node as state in the search algorithm
Constraint Propagation	19 second	Need to use pthread library on Linux kernel	The possibility of numbers is remove once the numbers is exist

Table 2.1 Research comparative Study

Apps Comparative Study

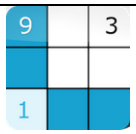
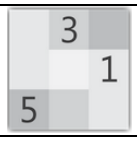
Apps / Details	Sudoku	Travel Sudoku	Sudoku Game	Sudoku Ninja	Sudoku Helper
Apps Icon					
Price	Free	Free	Free	Free	Free
Android required	1.6 and up	2.3 and up	2.3 and up	2.3 and up	2.3 and up
Level of difficulty	7 level	7 level	4 level	3 level	3 level
Background music	No	No	Yes	No	Yes
Generate answer scheme	Yes	Yes	Yes	Yes	Yes
Special Feature	Attractive interface	Has alarm system 17 clued sudoku puzzle (hardest difficulty level)	Provide Sudoku fact	Intuitive interface in high definition	Provide Hint

Table 2.2 Apps Comparative Study

CHAPTER 3

METHODOLOGY

While doing this Final Year Project, author had used several ways on how to get the finding and data. First of all, author have conducted an internet research and find five journals that are related to the Final Year Project's topic. To keep this report updated and fresh, author decided to select the journals that were written form years 2008 until now.

Besides, author planned to use Software Development Life Cycle method while developing the project. Author need to collect requirement first in order to develop the Sudoku Helper application. After that, author will design the prototype and implement the real application. Once the application is finished, the product will be move to the next stage which is testing phase. This phase involve a lot of testing and finding bug within the application. Finally, author will need to keep updating the application from time to time.

3.1 Research Methodology

The data gathering process will be done by questionnaire survey using Google Doc. The purpose of this survey is to get the requirement for this Final Year Project 1. It is also important to know what the users want and delivered it to them. The targeted participant for this survey is people who like to play Sudoku especially secondary school's students.

The survey will be done in two phases which is pre-development (FYP1) and post-development (FYP2). By having the result from this survey, further analysis can be done regarding the outcome of the system and how much users are willing to pay for the application.

3.2 Development Methodology

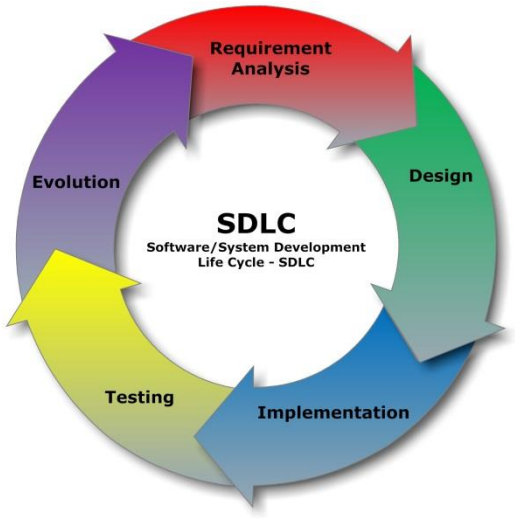


Figure 3.1. Software Development Life Cycle

Requirement Analysis

This is the first stage where author have to do research to know about previous case study on what software development have been done and what never been done regarding Sudoku application. From the internet research, author decided to do an application entitled 'Sudoku Helper' which is have more functions compared to previous developed application.

Design

The next phase is design stage. This is where the process of designing the application takes place. It is important to make a right Unified Modeling Language (UML) diagram such as use case diagram, sequence diagram, class diagram and many more. The design of Graphical User Interface (GUI) of this application also needs to be done in this period. It is to give author a big picture of how this project going to be and look likes.

Implementation

Implementation stage is the process where author start to develop the project by writing coding for that particular application. This process determine what software used to write the coding, how to transform a sketch of GUI into a real GUI and so on.

Testing

The main purpose of testing is to find bug and fix it. A bug can give drawback to a system and that is what every developer trying to avoid. If a system contains a lot of bug, it will affect all parts of the system and the system might not perform its function well.

3.3 System Interface

The application need to be user-friendly and provide ease of use to the user especially newbie or beginner player. The application is developed for mobile phone with Android operating system. All the programs is coded by using Android Studio software and author used Java Programming language for this system.

3.4 Tool Required

Research

i. Internet
ii. Laptop / Personal computer
iii. Journals
iv. Printer
v. Adobe Illustrator CS6

Development

i. Internet
ii. Laptop / Personal Computer

iii. Android Studio software
iv. Java Development Kit
v. Android SDK
vi. Android device (Mobile phone / Tablet)
vii. Google Play

CHAPTER 4

RESULT AND DISCUSSION

This chapter will discuss about the result of questionnaire and the expected outcome of this system. Author will explain by using UML diagram and Graphical User Interface (GUI) about what this system are supposed to do.

4.1 Result and Analysis of the Survey

4.1.1 Pre – development survey

A survey had been conducted to the targeted user by using Google Doc. The questionnaire basically based on age generation to see whether the respondents know the Sudoku game or not. It is also important to know which generations are familiar with mobile apps and they might be the potential buyer of this application later. The total respondent for this survey is 20 people. The result of this survey is illustrated as below.

Question 1

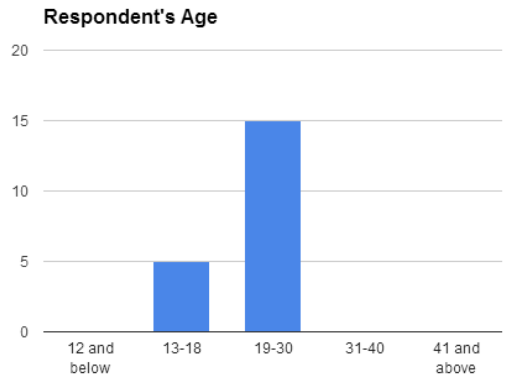


Figure 4.1 Respondent's Age

Out of 20 respondents, 15 of them are around 19 to 30 years old while another 5 are around 13-18 years old. It is showing that the people of 31 years old and above are not quite interested in playing games.

Question 2

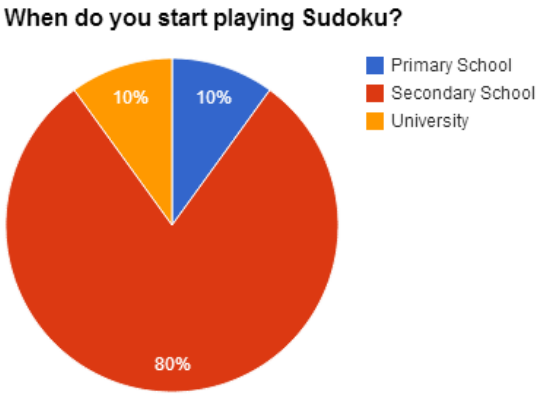


Figure 4.2 Respondent start playing Sudoku

In figure 4.2, most of the respondents started playing Sudoku while in secondary school. 2 of the respondents start playing Sudoku when in primary school and another 2 in their university time frame.

Question 3

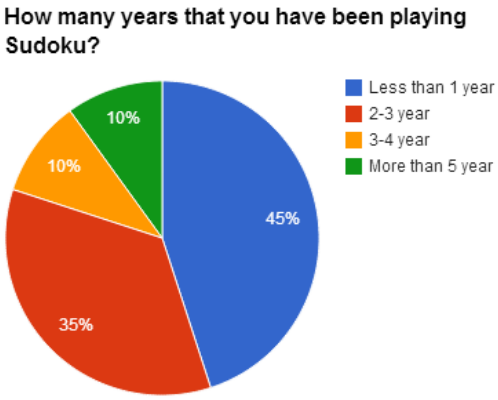


Figure 4.3 Respondent's experience

Based on the above graph, 45% of the respondent are considered beginner because they just played the Sudoku in less than a year. 35% of them had been playing Sudoku for about 2-3 years and 10% each for 3-4 years and more than 5 years' experience.

Question 4

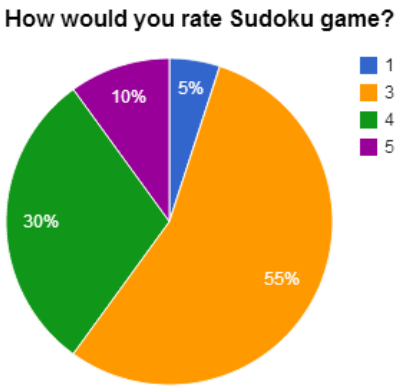


Figure 4.4 Respondent rating

This question is asking respondent's opinion on the Sudoku game itself whether it is interesting or not. Number 1 is the least interesting while number 5 is very interesting. Based on the result gathered, 55% say that Sudoku game is an average game. There also about 10% of the respondent who find this game interesting and 5% who not quite attracted to the game.

Question 5

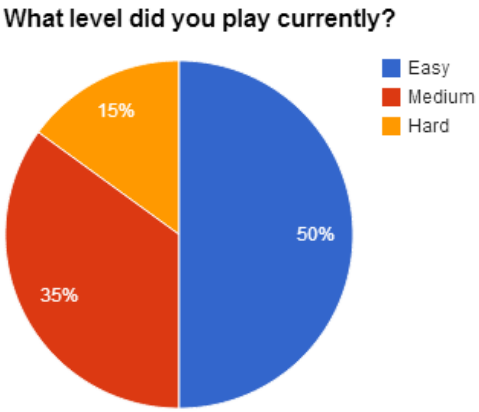


Figure 4.5 Difficulty level

Basically, there are 3 level of difficulty in playing Sudoku which are easy, medium and hard level. Since most of the respondents are new to the game, 50% of them are just playing easy level. On the other hand, 35% of them played medium level while the rest played the hard level.

Question 6

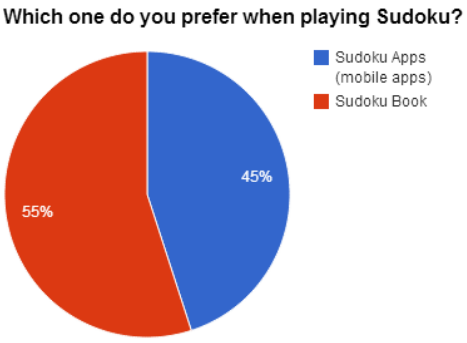


Figure 4.6 Respondent preference

Nowadays there are a lot of ways in playing Sudoku. 55% of the respondent prefer to use book while playing the game while 45% of the respondent will use mobile application when playing the game.

Question 7

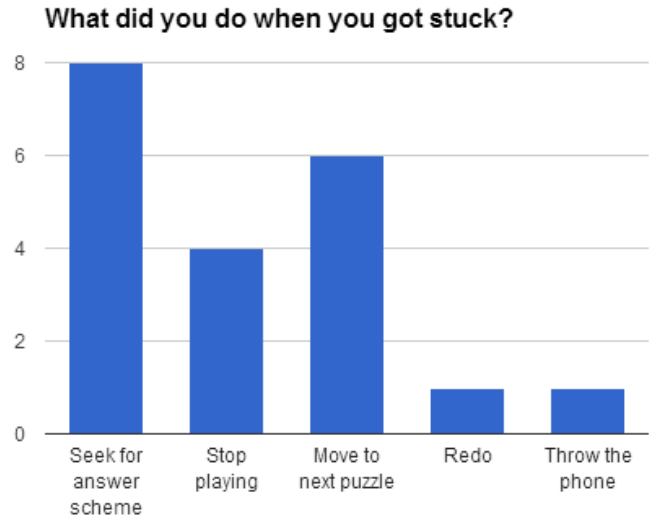


Figure 4.7 Respondent responds to stuck problem

Since Sudoku is a logic game, many of the players got stuck while playing it. Based on the survey, 8 respondents will look for answer scheme and 6 respondents will move to the next puzzle when they got stuck. There are also 4 respondents who will stop playing due to that problem. One respondent will try to redo the puzzle and another one tend to throw his/her phone if he/she cannot solve the puzzle.

Question 8

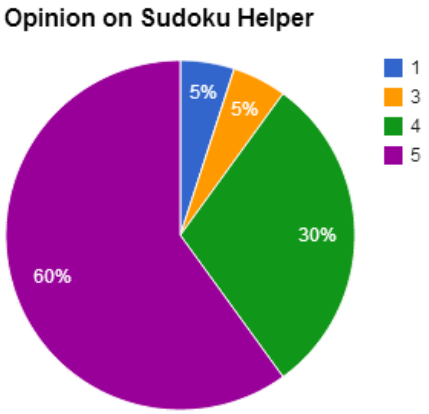


Figure 4.8 Respondent's opinion on Sudoku Helper

This question is about asking respondents' opinion on Sudoku Helper, a mobile application which can provide hint when user needs it. From five scales which is 1 is a bad idea and five is a great idea, 60% of the respondent agree that to would like to use this application.

Question 9

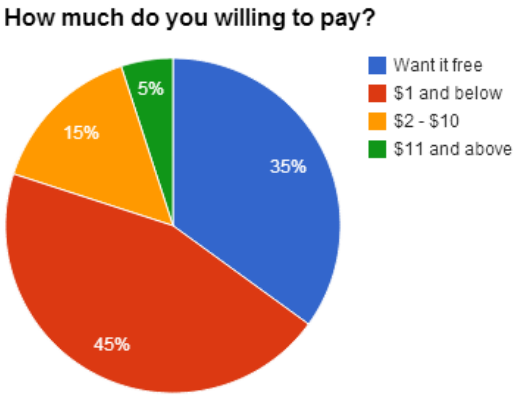


Figure 4.9 Respondent willingness to pay

Based on above pie chart, 45% of the respondent are willing to pay for less than $1 while another 35% of them want the application free. 15% are willing to pay for the price range from $2 to $10 and 5% can pay for $11 and above.

Question 10

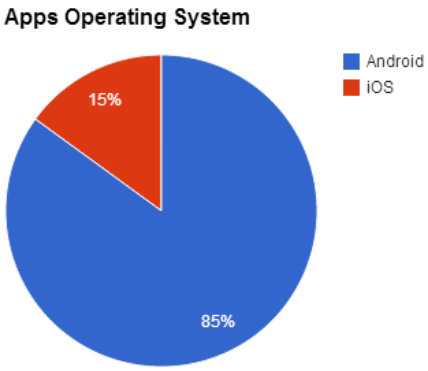


Figure 4.10 Apps operating system

On the final question of the survey, 85% of the respondent would prefer to use android operating system while the rest like to use iOS operating system.

4.1.2 Post-Development Survey (User Acceptance Test)

The system had been tested by 10 people who live around Seri Iskandar area. The system is tested under the instruction of the developer and the developer take note about some comments regarding the application. After testing with the prototype, the testers are asked to fill in the survey form to get their feedback. The result of the user acceptance testing will be discussed through figure below.

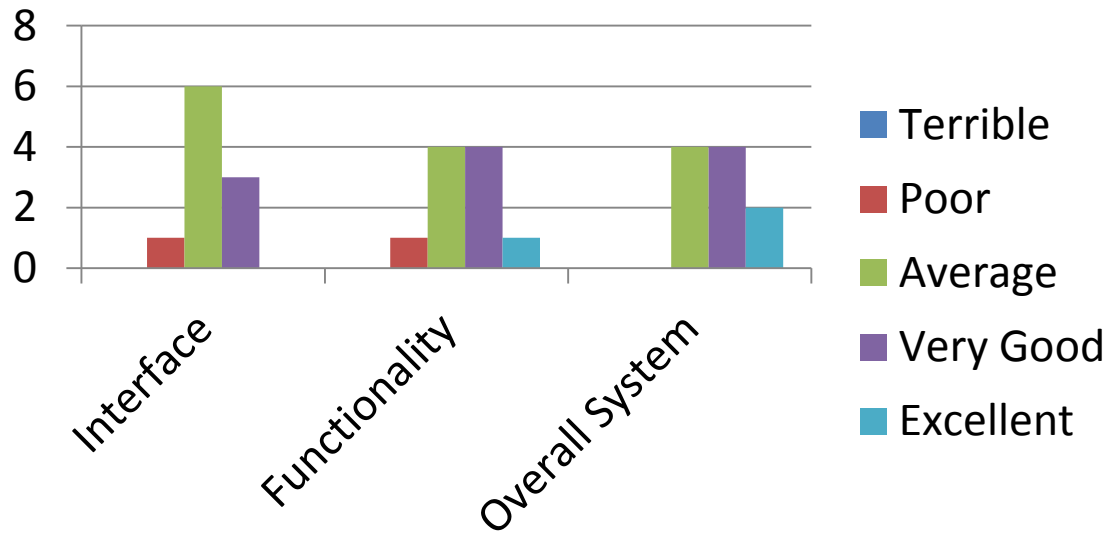


Figure 4.11 User Acceptance Testing data

4.2 Use Case Diagram

A use case is a specification of sequences of actions, including variant sequences and error sequences, that a system, subsystem, or class can perform by interacting with outside actors [9]. Use case typically represents a major piece of functionality that is complete from beginning to end and captures a contract between the stakeholders of a system about its behavior [10]. This use case diagram will be divided into several actions such as main menu, start game, how to play the game and lastly quit the application.

4.2.1 Main Menu

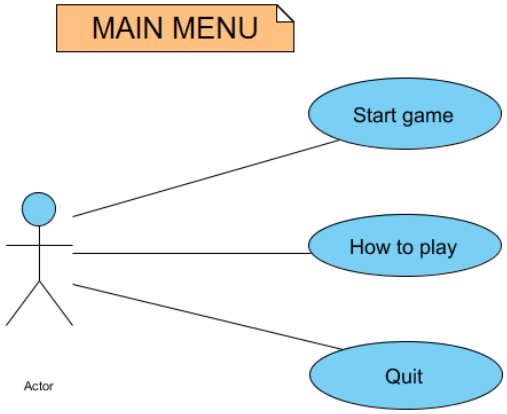


Figure 4.12 Use case diagram for main menu

Purpose	To show the option of this application.	
Actor	User	
Trigger	User can choose whether to play a Sudoku game, to read the Sudoku game rules and regulations or quit the application.	
Precondition	The user must have an Android operating system. The user must install the Sudoku Helper application.	
Scenario Name	**Step**	**Action**
	1	Open the Sudoku Helper application.
	2	Choose one of the options given on the interface.

4.2.2 **Start Game**

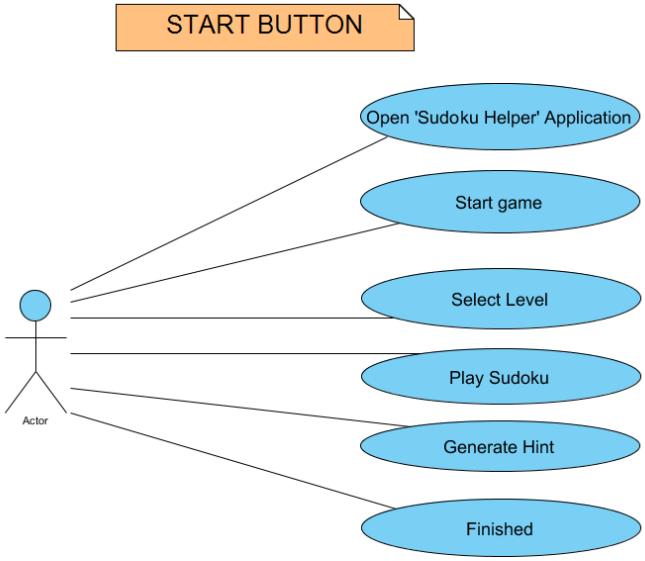


Figure 4.13. Use case for start game

Purpose	To allow user to play Sudoku game.	
Actor	User	
Trigger	Click 'Start' button to play and solve Sudoku puzzle .	
Precondition	The user must have an Android operating system. The user must install the Sudoku Helper application.	
Scenario Name	**Step**	**Action**
	1	Open the Sudoku Helper application.
	2	Choose 'Start' button to start playing the game.
	3	User need to choose the level of difficulty of the game
	4	User can start playing by fill in the answer in the blank space
	5	User can generate a hint if they got stuck
	6	Once completed the puzzle, user need to submit the answer to check with the answer scheme.

4.2.3. How To Play

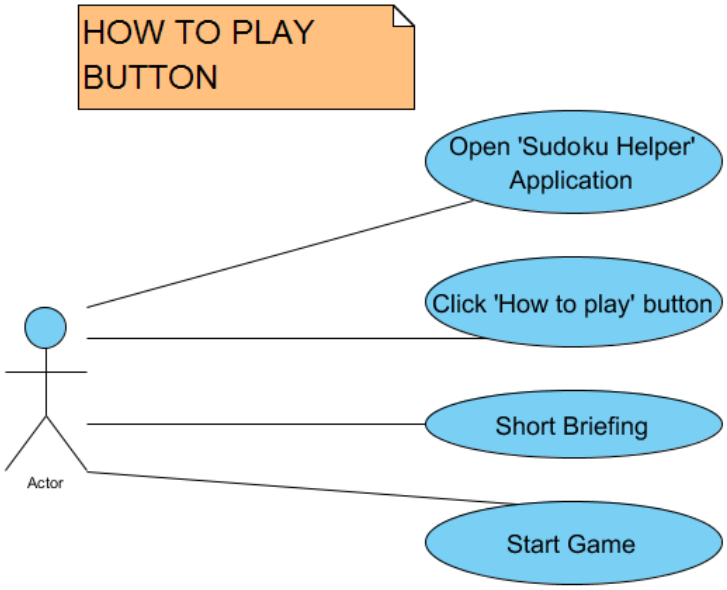


Figure 4.14. Use case for how to play

Purpose	To show the rules and regulations for newbie player.	
Actor	User	
Trigger	Click ‘How to play’ button to view the rules and regulations.	
Precondition	The user must have an Android operating system. The user must install the Sudoku Helper application.	
Scenario Name	**Step**	**Action**
	1	Open the Sudoku Helper application.
	2	Choose ‘How to play’ button.
	3	A user will be given a short briefing on the rules and regulations of Sudoku game.
	4	User then will be allow to start playing the game.

4.2.4 Quit Application

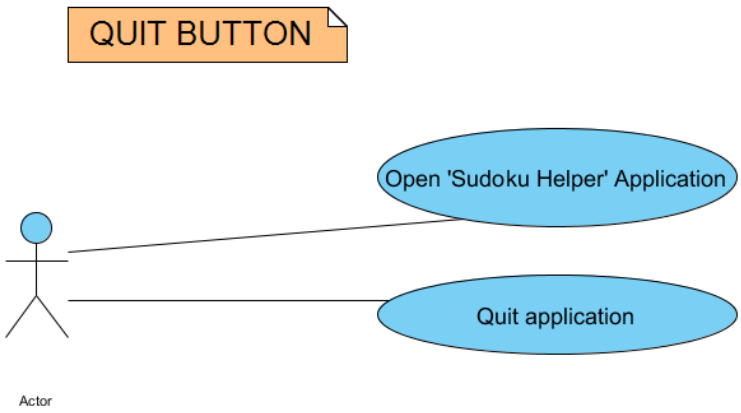


Figure 4.15. Use case for quit application

Purpose	To close the Sudoku Helper application.	
Actor	User	
Trigger	Click ‘Quit’ button.	
Precondition	The user must have an Android operating system. The user must install the Sudoku Helper application.	
Scenario Name	**Step**	**Action**
	1	Open the Sudoku Helper application.
	2	Close and quit the application.

4.3 Activity Diagram

Activity diagram is graphical representation of workflow of activities and actions with support for choice, iteration and concurrency. The activity diagram show the overall of the flow in that particular system [11].

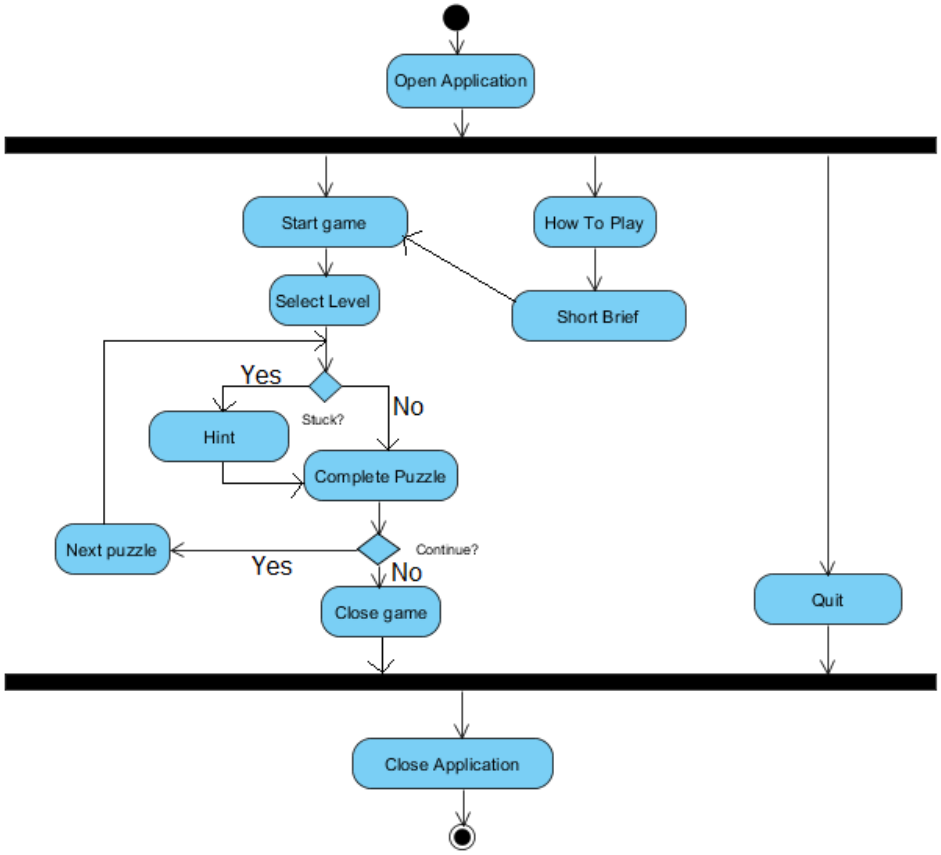


Figure 4.16 Activity diagram of Sudoku Helper application

First of all, every user must open the Sudoku Helper application first in order to play the game. Once the application is already running, user will be given 3 options whether to immediately playing the game, read the rules and quit the application.

The first option is playing a Sudoku game. If a user decided to play the game, user will be directed to the game interface of the system. A user may select the level of difficulty of the game and can start playing the game. If user got stuck while playing the game, this application can generate a hint for user to proceed with the next step. Once completed the puzzle, user are given 2 options whether to continue to next puzzle or close the application.

Back to the main menu option again, the second choice that a user can select is to read the rules and regulations of the game. The application will display simple rules so that it is easier for the beginner player to understand it. After finish read all the rules, user then can start playing Sudoku game.

The third and last option is to close and quit the application. Once user click select 'Quit' button, the system may and may not save the current game based on user preference and exit the application.

4.4 Graphical User Interface

Figure 4.17 Main menu interface

Figure 4.18 Select level interface

Figure 4.19 Game interface

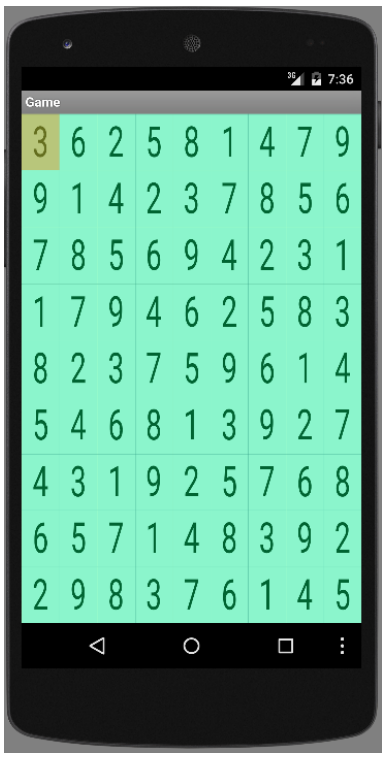


Figure 4.20 Game Completed interface

Figure 4.21 Rules and regulations interface

CHAPTER 5

CONCLUSION & RECOMMENDATION

The Sudoku Helper is an android application intended to give fun for the Sudoku lover from all over the world. The algorithm used can generate hint faster and it will give satisfaction for the users. In addition, this application is simple to use and it has a very nice interface.

However, there are some improvement and features that can be added in the future to make the application more reliable and attractive. A few examples of the features are as follows:

i. MP3 player
ii. Counting Watch
iii. Alarm

REFERENCES

[1] (2015, February 10) Sudoku. Retrieved from http://en.wikipedia.org/wiki/Sudoku

[2] Gary, M., Bastian, T., Gilles, C. (2013). Introduction. *There is no 16-Clue Sudoku: Solving the Sudoku Minimum Number of Clues Problem via Hitting Set Enumeration*

[3] Harrison, C., Malcolm, E.R. (2012). Paclets and Human Solving Symmetry. *A Group-theoretic Approach to Human Solving Strategies in Sudoku*, 3(3)

[4] Gary, M., Bastian, T., Gilles, C. (2013). Overall Strategies of Checker . *There is no 16-Clue Sudoku: Solving the Sudoku Minimum Number of Clues Problem via Hitting Set Enumeration.*

[5] Urszula, B., Przemyslaw, J. (2012). The Differential Evolution Algorithm. *Solving the Sudoku with Differential Evolution*, 9

[6] Eremic, M. et al (2013). Comparison of State Space Search Algorithm – Sudoku Puzzle Game. *A Journal for Information Technology, Education Development and Teaching Methods of Technical and Natural Sciences*, 3(1)

[7] Alton, C., Ehsan, N., Rafat, R. (2012). Constraint Propagation. *Parallelization of Sudoku.*

[8] "comparative study." A Dictionary of Nursing. 2008. Retrieved March 29, 2015 from Encyclopedia.com: http://www.encyclopedia.com/doc/1O62-comparativestudy.html

[9] Rumbaugh, J., I. Jacobson, et al. (1999). The Unified Modeling Language Reference Manual. Reading, Addison Wesley.

[10] Cockburn, A. (2000). Writing Effective Use Cases. Reading, Massachusetts, Addison-Wesley.

[11] (2015, March 27) Activity Diagram. Retrieved from http://en.wikipedia.org/wiki/Activity_diagram

APPENDIX 1

Gantt Chart FYP

No	Details/Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28
	Planning Phase																												
1	Research on topic																												
2	Identify problem and propose solution																												
3	Clarify scope, goals and objectives																												
4	Prepare system requirement																												
5	Comparative Study																												
6	Confirmation of FYP topic																												
7	Conduct Online Survey																												
8	Analyze survey data																												
	Design Phase																												

9	Design graphical user interface																												
10	Do the UML diagram																												
	Implementation Phase																												
11	Coding the program																												
12	Implement the algorithm																												
	Testing Phase																												
13	Usability testing																												
14	Functionality Quality Assurance																												
15	User acceptance testing																												
	Launch Phase																												
16	Viva																												
17	Project Documentation																												
18	Launch the system																												

APPENDIX 2

QUESTIONNAIRE

Sudoku Helper Apps

*Make sure you have play Sudoku before answering the questionnaire !!

* Required

How old are you? *

- ○ 12 and below
- ○ 13-18
- ○ 19-30
- ○ 31-40
- ○ 41 and above

When do you start playing Sudoku? *

- ○ Primary School
- ○ Secondary School
- ○ University
- ○ Work place
- ○ Other: _____

How many years that you had been playing Sudoku? *

- ○ Less than 1 year
- ○ 2-3 year
- ○ 3-4 year
- ○ More than 5 year

How would you rate Sudoku game? *

	1	2	3	4	5	
Not interesting	○	○	○	○	○	Very interesting

What level did you play currently? *

- ○ Easy
- ○ Medium
- ○ Hard

Which one do you prefer when playing Sudoku? *

- Sudoku Book
- Sudoku Online (website)
- Sudoku Apps (mobile apps)

When you got stuck while playing Sudoku, what did you do? *

- Seek for answer scheme
- Stop playing
- Move to next puzzle
- Other: [ ]

What are your opinion on apps that can provide hint when player got stuck? *

	1	2	3	4	5	
Bad	○	○	○	○	○	Great

How many ringgit are you willing to pay for the apps? *

- Want it free
- $1 and below
- $2 - $10
- $11 and above

Do you prefer apps on Android or iOS? *

- Android
- iOS