

CHAPTER ONE

INTRODUCTION

1.1 Background of the Project

Scrabble is an old classic board game, with game rules that are not very complicated. Compared to other classic board games, such as Chess, there are more factors to take into consideration when making a move in Scrabble. Apart from anagramming and generating words, there are other crucial decisions to make. A player would probably find several legal moves in one round, and then would have to decide which one to place. Choosing the move that generates the highest score is not always the ideal way to go. It could be that making such a move would create a situation in the next round where nothing could be done, or let the opponent score highly.

Making the choice of which move to use include many factors, and there are several different techniques and strategies one can follow to make the decision easier. Hitting the bonus squares can generate an extremely high score, hence one strategy is to try to always hit the bonus squares and prevent the opponent from doing the same. Another way of being successful is to prioritize using letters with high score, since they would not only generate a high total score to the player, but could also create difficult situations in the future if not used as soon as possible.

Scrabble game is a simple game played by four, three, two even one person. Scrabble game is also played with tiles and encountered on a special and well-designed board. The game can be played with a board disserted with a cross-like path. In between any of the two arms of the cross path is a center square calved out as a square box, which has a symbol of star.

At the down side of this board, there is a device called a Rack where the tiles are been placed before they are picked. This rack can be any colour depending on the developer or the user, titles and the rack can have the same colour. This rack is occupied by seven tiles at a starting and can be change or shuffle when the need arises. Each player has a starting point (middle) where he/she will place the tiles when picked. This spot is of different colour as the box (others) and is called the starting point. Other features of the board include smaller square boxes that make up the steps on the path which each tile steps on as it enters the path to produce a word. There are double and tripple word squares on the board. In the double word lines or square if two tiles that will generate a meaning are placed there, the player will score a high mark likewise the tripple word square. There are tiles, which are used to call for placement, which determine how the score is been ranked on the score device. Some makes use of the standard set number, while some make use of only seven (7) depending on the players.

When the game is being played, the objective of each player is to navigate his tile on the path to achieve a meaningful word. And any player whose tiles finished first wins the games or any player that scores the highest point wins the game i.e. top score or best player. But in other hand, the player whose tiles remain after number-of- players have finish, come last and become the worst loser. Each player must abide by the

rules and regulation of the game that will be specified in subsequent chapters.

1.2 Statement of the Problem

Because scrabble game is played manually, young people do not have much interest in it. But making it computer based will make young people to enjoy it.

1.3 Objective of the Project

The objective of this study is stated as following:

To make scrabble game join the most other games in computer “world”

To handle cheating in scrabble game and to standardize the game.

To simulate scrabble game using computer.

1.4 Project justification

This project is justified because making scrabble computer based enhances learning.

1.5 Scope of the Project

The scope of this project is to design a computer based scrabble game.

1.6 Project Report Organization

This project is organized into six chapters. Chapter one is the Introduction, chapter two is the Literature Review, chapter three is the Methodology and System Analysis, chapter four is the System Design, chapter five is the System Implementation and Testing and chapter six is the Summary, Conclusion and Recommendation.

1.7 Definition of terms

Anchor square: Vacant board square adjacent to a placed tile.

Bag: Tiles that have not been drawn by players.

BOR: Balance on rack player.

BS: Bonus square player.

Cross check set: Set of possible letters on an anchor square with regard to adjacent word going vertical from named square.

DAWG: Directed acyclic word graph.

HSW: High score word player.

Vowel ratio: Ratio describing how many of the total number of tiles are vowels.

Game: game is any activity which is executed only for pleasure and without conscious purpose

Programming: The act of instructing a computer to do your bidding

Center star Square: This is a start square n the board that doubles the word.

Database: Organized store computerized data.

Entertainments: This is a condition that is considered favourable

Exchange: The act of taken and replacing.

Hints: Piece of practical formation.

Human: This is a type of game choose in scrabble

I spy: This is a game played by order children.

Kits: Instruments used in playing scrabble game.

Play Now Button: A command button that make the game to start.

Premium Square: This is a start up square, which double the words.

Quit Button: A button used to terminate the operation of the game.

Rack: A device that holds the tiles in a scrabble game.

Score: To win or obtain a point in a competitive activity.

Scrabble Game: A game played on a board covered in square in which player wins point by combining letter.

Scrabble Dictionary: A device in the scrabble game where words are listed

Shuffle: Often to give an appearance of activity when nothing useful is being done.

Solitaire: One persons game e.g. word game.

Submit: This makes the letter to be permanent on the board and also score the player.

Tiles: A piece of plastic used in playing game.

Triple word score: This triples the number of word on it.

Triple letter score: This triples the number of letter placed on it.

CHAPTER TWO

LITERATURE REVIEW

In 1938, American architect Alfred Mosher Butts created the game as a variation on an earlier word game he invented called Lexiko. The new game, which he called "Criss-Crosswords," added the 15×15 gameboard and the crossword-style game play. He manufactured a few sets himself, but was not successful in selling the game to any major game manufacturers of the day.

In 1948, James Brunot, a resident of Newtown, Connecticut – and one of the few owners of the original Criss-Crosswords game – bought the rights to manufacture the game in exchange for granting Butts a royalty on every unit sold. Though he left most of the game (including the distribution of letters) unchanged, Brunot slightly rearranged the "premium" squares of the board and simplified the rules; he also changed the name of the game to "Scrabble", a real word which means "to scratch frantically". In 1949, Brunot and his family made sets in a converted former schoolhouse in Dodgingtown, a section of Newtown. They made 2,400 sets that year, but lost money. According to legend, Scrabble's big break came in 1952 when Jack Straus, president of Macy's, played the game on vacation. Upon returning from vacation, he was surprised to find that his store did not carry the game. He placed a large order and within a year, "everyone had to have one.

In 1952, unable to meet demand himself, Brunot sold manufacturing rights to Long Island-based Selchow and Righter, one of the manufacturers who, like Parker Brothers and Milton Bradley Company, had previously rejected the game. In its second year as a Selchow and Righter-built product, nearly four million sets were sold.

Selchow and Righter bought the trademark to the game in 1972. JW Spears began selling the game in Australia and the UK on January 19, 1955. The company is now a subsidiary of Mattel. In 1986, Selchow and Righter was sold to Coleco, who soon after went bankrupt. The company's assets, including Scrabble and Parcheesi, were purchased by Hasbro.

In 1984, Scrabble was turned into a daytime game show on NBC. Scrabble ran from July 1984 to March 1990, with a second run from January to June 1993. The show was hosted by Chuck Woolery. The tagline of the show in promo broadcasts was, "Every man dies; not every man truly Scrabbles." In 2011, a new TV variation of Scrabble, called Scrabble Showdown, aired on The Hub cable channel, which is a joint venture of Discovery Communications, Inc. and Hasbro. Scrabble was inducted into the National Toy Hall of Fame in 2004

2.1.1 Dictionary representation

Words from a dictionary can be stored by representing the letters as edges in a trie. A path represents a word. Each path can have sub-paths that

represent shorter word variations by letting the nodes mark the end of a word.

Appel and Jacobsen showed that the size of the dictionary representation can be reduced with a Directed Acyclic Word Graph, referred to as a DAWG, Jacobsen, (1988). The DAWG can be constructed by first creating a trie and minimizing it by finding cases where two or more words can share a common letter (node).

A new edge is then created from the previous node in one of the words to the other words node. Finally, the unnecessary edge and node are removed.

A trie has a lot of redundancy, because identical edges and nodes are stored, while the DAWG allows elimination of duplications.

A dictionary consisting of more than 100 000 words occupies 0.5 MB with a trie, but is reduced to only 175 kB with a DAWG

2.1.2 Word generation algorithm

Finding and forming words to lay out on the board is difficult. Appel and Jacobsen presented a solution to the problem by reducing the board to one dimension. Instead of searching both horizontally and vertically, it is possible to generate words only horizontally. The argument is that generating a word vertically is basically the same thing, with the only difference that the board is transposed. Therefore, the word generation algorithm is limited to only finding words horizontally. Two searches are made for each move, and one of them over the transposed board Jacobsen, (1988).

Anchor squares

A key in the algorithm implemented is the concept of anchor squares. Anchor squares are empty squares adjacent to the words placed on the

board. These are important since words can only be extended from already existing tiles. In the first move of the game there is only one anchor; the center square, since the word in the first turn always has to be placed over the center square.

The adjacent tiles are the anchor tiles from which the word generation begins.

Cross-checks

The set of available legal letters for one square is in Appel and Jacobsen's paper, Jacobsen, (1988). referred to as a cross-check set. When placing a word horizontally, the vertically placed letters also have to form a legal word. It is quite easy to establish that if a word is placed horizontally, the vertical word can increase with only one letter at a time. This makes it possible to calculate, for each anchor square, which set of letters that are possible to place at that square.

The calculations can be made before each move, and allows the algorithm to place a word by row, without considering the rest of the board.

2.1.3 Game strategies

Some of the different Scrabble game strategies are:

Difficult letters: letters with high score are usually more difficult to place and should be used as soon as possible e.g. Z or C.

Balance on the rack: it is easy to get stuck with only consonants or only vowels on the rack. Keeping a good balance can be a good idea to help construct words in the next turn.

Bonus squares: hitting the bonus squares will generate a higher score. It is also important to occupy them first to destroy the bonus opportunities for the opponent.

Word extensions: the ability to identify extensions (i.e. prefixes and suffixes) to words, is a key to being a successful player. Such moves can

generate very high scores, as the word will use many existing tiles, Sheppard, (2002).

Vocabulary: there are many short words in the language, and it is rewarding if a player can learn many short word letters which are easy to lay out.

2.2 Theoretical Review

2.2.1 Review of Simulation Model

Simulation, according to Carson (2000), is “the process of designing a model of a real system and conducting experiments with this model for the purpose either of understanding the behavior of the system or of evaluating various strategies (within the limits imposed by a criterion or set of criteria) for the operation of the system.” According to this definition, a simulation can be a discrete-event simulation, as we will discuss in this project. The difference, and the power, of discrete-event simulation is the ability to mimic the dynamics of a real system, Law and Kelton (2000). Many models, including high-powered optimization models, cannot take into account the dynamics of a real system. It is the ability to mimic the dynamics of the real system that gives discrete-event simulation its structure, its function, and its unique way to analyze results.

Discrete-event simulation: Has evolved a basic structure that is used by most simulation packages. Regardless of how complex a discrete-event simulation package may be, it is likely to contain the basic components that we will describe in this section. The structural components of a

discrete-event simulation include entities, activities and events, resources, global variables, a random number generator, a calendar, system state variables and statistics collectors.

2.2.2 Entities

The best way to understand the function of an entity is to understand that entities cause changes in the state of the simulation, Banks, et. al. (2000). Without entities, nothing would happen in a simulation. As a matter of fact, one stopping condition for a simulation model is the condition where there are no active entities in the system. Entities have attributes. Attributes are characteristics of a given entity that are unique to that entity. Attributes are critical to the understanding of the performance and function of entities in the simulation.

2.2.3 Activities and Events

Activities are processes and logic in the simulation. Events are conditions that occur at a point in time which cause a change in the state of the system. An entity interacts with activities. Entities interacting with activities create events, Ingalls and Kasales (1999).

There are three major types of activities in a simulation: delays, queues and logic. The delay activity is when the entity is delayed for a definite period of time. In general, the length of time for a delay is either constant or is randomly generated. At the point that the entity starts the delay, an event occurs. This event schedules the entity on the calendar (which we will get to later). If the delay is for d time units, then the entity is scheduled to complete the delay d time units after the current time of the simulation. At that time, the delay expires and another event is generated. Queues are places in the simulation where entities wait for an unspecified period of time. Entities can be waiting on resources (which we will get to later) to be available or for a given system condition to occur. Queues are

most commonly used for waiting in line for a resource or storing material that will be taken out of the queue when the right conditions exist.

Logic activities simply allow the entity to effect the state of the system through the manipulation of state variables (which we will get to later) or decision logic.

2.2.4 Resources

In a simulation, resources represent anything that has a restricted (or constrained) capacity. Common examples of resources include workers, machines, nodes in a communication network, traffic intersections, etc. Depending on the target market of the simulation package, many complex resources are available for use.

2.2.5 Global Variables

If you are a programmer, then the idea of having global variables is nothing new. A global variable is a variable that is available to the entire model at all times. A global variable can track just about anything that is of interest to the entire simulation. In our model we have four global variables, two of which help us configure the problem and two of which collect revenue information. The two variables that help us configure the problem are the length of the line allowed.

2.2.6 Random Number Generator

Every simulation package has a random number generator. The random number generator (technically called a pseudo-random number generator) is a software routine that generates a random number between 0 and 1 that is used in sampling random distributions. For example, let us assume that you have determined that a given process delay is uniformly

distributed between 10 minutes and 20 minutes. Then every time an entity went through that process, the random number generator would generate a number between 0 and 1 and evaluate the uniform distribution formula that has a minimum of 10 and a maximum of 20. As an example, let us assume that the generated random number is 0.7312, then the delay would be $10+(0.7312)*(20-10) = 17.312$. So the entity would delay for 17.312 time units in the simulation. Everything that is random in the simulation uses the random number generator as an input to determine values.

2.2.7 The Calendar

The calendar for the simulation is a list of events that are scheduled to occur in the future. In every simulation, there is only one calendar of future events and it is ordered by the earliest scheduled time first. In a later example, it will become more clear how the calendar works and why it is important in the simulation. At this point, just remember that, at any given point in time, every event that has already been scheduled to occur in the future is held on the calendar.

2.2.8 System State Variables

Depending on the simulation package, there can be several system state variables, but the one system state variable that every simulation package has is the current time of the simulation. In order to keep from offending any simulation vendors, we will choose a different name for our simulation time variable. Our name of the current time of the simulation is `CurrentTime`. This variable is updated every time an entity is taken from the calendar.

2.2.9 Statistics Collectors

Statistics collectors are a part of the simulation that collects statistics on certain states (such as the state of a resources), or the value of global variables, or certain performance statistics based on attributes of the entity. There are three different types of statistics that are collected, counts, time persistent, and tallies. Counts are very straightforward, they count. In our model, we count the number of Lost Customers because the line was too long. Time-persistent statistical collectors give the time-weighted values of different variables in the simulation. A common variable to track is the utilization of a resource.

2.3 History of Games

Games especially card games and board games, are as old as civilization itself Egyptian art carried on tombs about 2500 BC showed board games played. The ancient Egyptians also played games with balls. Indian tribes of Central America have ball game long before Christopher Columbus arrived in the new world.

The Olympic Games ranked among the most famous sports contents up to ancient times. The Greeks held game as early as 776 BC in honor of the god zero. Popular children's games in ancient Rome included bowling hoops, blind man buff and tug-of-war, chess originated in Asia and spread to westwards to Persia during the A.D 500's. It then spread to Arabia and Spain and throughout Europe.

2.4 Three Basic Types Of Game

As a result of research that was carried out during the development this project, a lot of books and materials were combined such as the world Book Encyclopedia (W B E P₂₂) which states clearly that there are three types (basic) of games via.

- Lucky games
- Games that involves skill
- Physical Endurance Game

Board game such as scrabble game is an example of skill game since the players success depends on the outcome of scores on the placement of tiles. The players must have skill to win games such as darts and rings toss. However, many games call for the combination of luck, endurance and skills.

2.4.1 Games for Young Children

Children enjoy just like adults do. The world Book Encyclopedia specifies the following children's game that only provides enjoyment for young children, but also helps develop physical skills and sharp observation. These game enable the players (children) learn to adjust to new situation and to follow rules (WBE P₂₂)

2.4.2 Jump Rope Game

This is a type of game that is mostly played by young girls but many boys have also turns to play, developing the skill to jump one or two ropes at a

time. Two players turn the rope while chanting a rhyme. After the player finishes the chant, he or she move away from the turning rope. Another player moves in and started to chant while jumping the rope. A player who fails to jump the rope must drop out of the game.

2.4.3 Games for Older Children

Since there is advancement in things of the world, so there is advancement in Children. As children grow older, they prefer more challenging games that test their skill. Among the games that have been long in favour among older children are the following two games:

- **Blackboard Relay**

This is a game that is been developed in form of a boardgame. This game called blackboard relay is a classroom game played by two teams. On a signal from the teacher, a player from each team writes a word on the blackboard. Then the players return to their mates and then give chalk to the next player to write a complete sentence.

- **ISPY**

This is also known as “Twenty questions on animals, vegetables, or minerals. One player thinks of an object and say, “ISPY” if the object is not nearby, he or she must give a clue to its location. The other players try to guess the object by asking questions. But the first player

can answer only with “yes” or “no”. This first player wins the games, if others could not guess the right answer after twenty questions.

2.4.4 Games for adults

As children grew up, they begin to select the type of games that are suitable for them. Adults mostly have interest in some advanced games such as board games that required thinking and reasoning. Others enjoy outdoors games and sports, or indoor action games, and also games that are in conjunction with actions and thought. Word games and some guessing games become popular because they challenge imagination.

In scrabble for instance, players form meaningful words by combining blocks that have letters printed on them, players also reason wisely before making a decision or before playing the game, puzzle game and riddle are also popular. In a game called wrong capital a player purposely name a capital in a wrong state. His opponent must locate the capital correctly. Board game played with movable pieces challenge the player’s foresight and imagination. These games include checkers chess, monopoly, life ludo game etc. In monopoly, players try to acquire more wealth and properties than their opponents by gaining monopolies in properties. In life, the players try to gain as much wealth as possible through investment and business deeds.

CHAPTER THREE

METHODOLOGY AND SYSTEM ANALYSIS

Methodology is the systematic, theoretical analysis of the methods applied to a field of study, or the theoretical analysis of the body of methods and principles associated with a branch of knowledge.

3.1 General Analysis of the System

Before advancing into designing of the scrabble game, serious feasibility studies of the manual system were carried out.

In manual play, each player keeps score, with a running tally, and these are compared from time to time throughout the game. Some players like to make these comparisons upon every play. This is found to be annoying and distracting. The going-out scoring in manual play is slightly different from the rules, which come with Scrabble sets. When the bag of tiles is empty and a player clears out his/her rack, the game is over and that player who just went out doubles that value of the tiles in the opponent's rack. Manual scrabble is played using tiles, when it is a player's turn, he shuffles the tiles in the bag or any material used and placed or submit on the board. If a tile is selected it must not be place on any other square except on the center start square which is pink, and always double the score for the first word. The game end when all possible play have been made, or when each player's scores zero points (by passing, exchanging or losing challenges) the three consecutive turns.

3.2 Main Menu Design

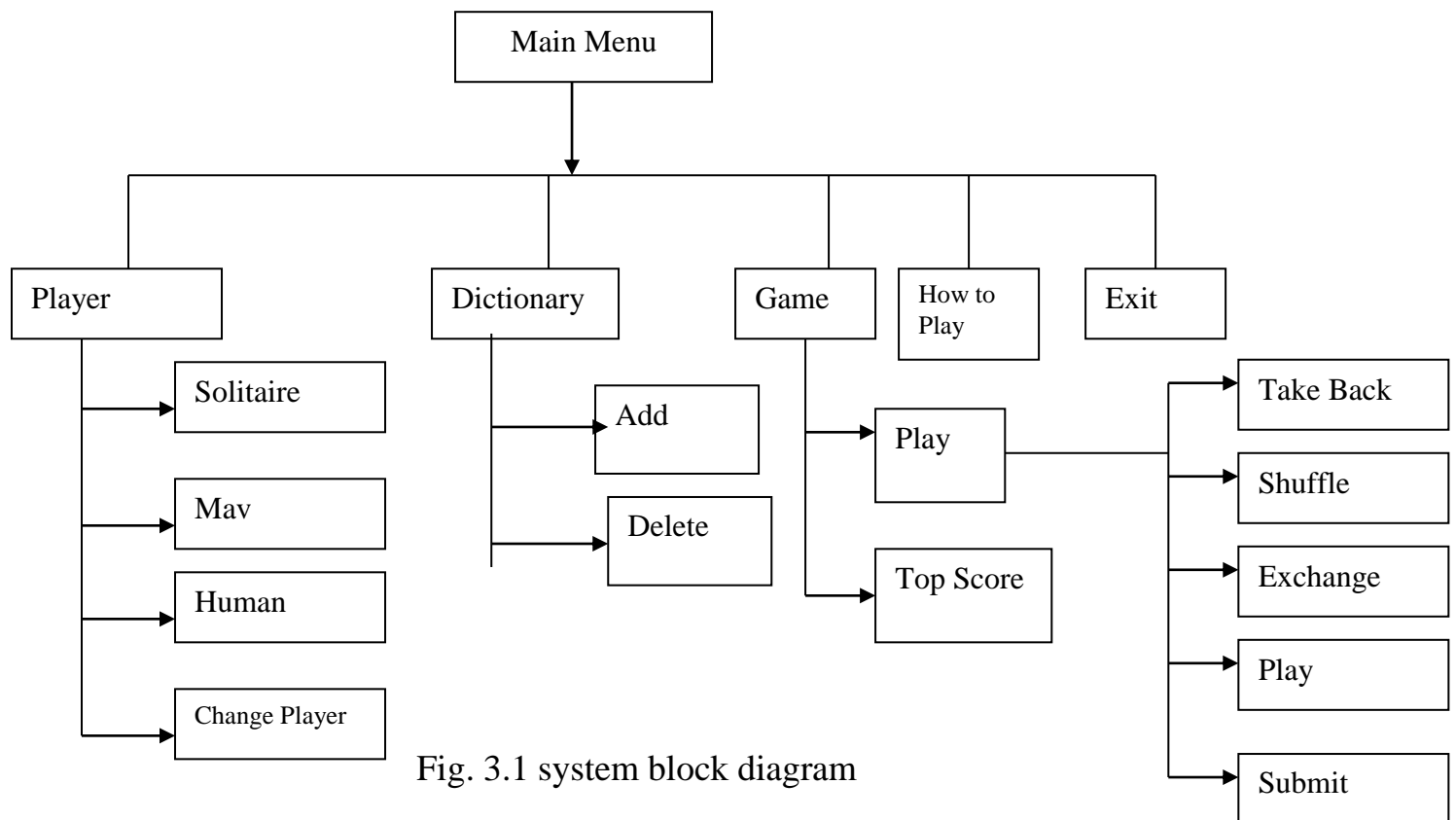


Fig. 3.1 system block diagram

3.3 Tools: The software used to develop this project are Microsoft Access 97 and Visual basic 6.0 and the hardware used is Microcomputer in general.

3.4 Method of Data Collection

- ❖ **Oral Interview:** - Many expert in scrabble game were interviewed and the result got from them were analyzed and used in this project.
- ❖ **Internet:** - Most data were collected from the internet. Sites such as National Scrabble Association; were visited for information on scrabble game.

3.5 Data Flow Diagram

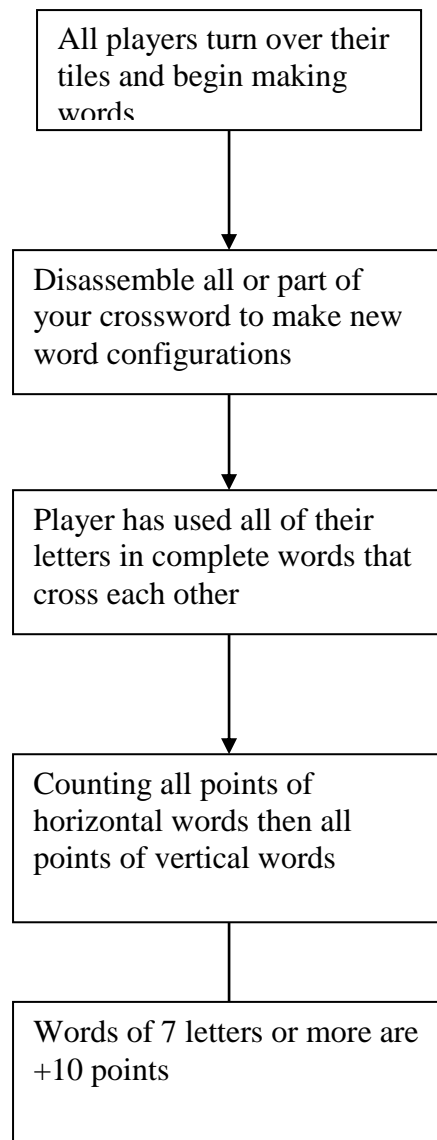


Fig 3.2, Data flow diagram

These are the various steps players will follow to start and end the game. The players will keep their tiles ready for play, after which, they clear the board to start forming new words. After using all the tiles they have in their bags, they start to count points horizontally, then vertically to know who is the winner. Any word that has seven letters or more has ten points.

3.6 Limitations of the present system

Some of the problems identified in the present system include:

- (a) Since the current system exists as a game that is played manually, people are abandoning the game for more advanced games that are played using computer
- (b) Games played manually are not enjoyable.
- (c) There is no automatic means of determining the winner apart from counting the tiles which may result in erroneous counting and misjudgment.

3.7 Analysis of the Proposed System

The proposed system is computer based. Implementing it will help to eradicate errors in scrabble game.

CHAPTER FOUR

SYSTEM DESIGN

4.1 Objective of Design

The system design aimed at developing software that can be used by people to play scrabble game. The objective of the design includes:

- Design a software that simulates scrabble game
- Design a well formatted graphic interface for the playing of the game
- Maintain a two player or more setting for the game
- Scoring of players
- Determining the winner at the end of the games

4.2 Output Specification and Design

The output specification and design of the scrabble game is in a graphical format. The scrabble board is designed with a form to show all the boxes for the game. The five colours used represents three different score letters like triple word score, triple letter score, double word score, double letter score and ordinary square box. The tiles played by the user are displayed on this board. At the end, the player scores a point, which is displayed at the scoreboard. The list of players also forms another output of the system with the highest scorer coming first on the list. Below is the output design of the scrabble game.

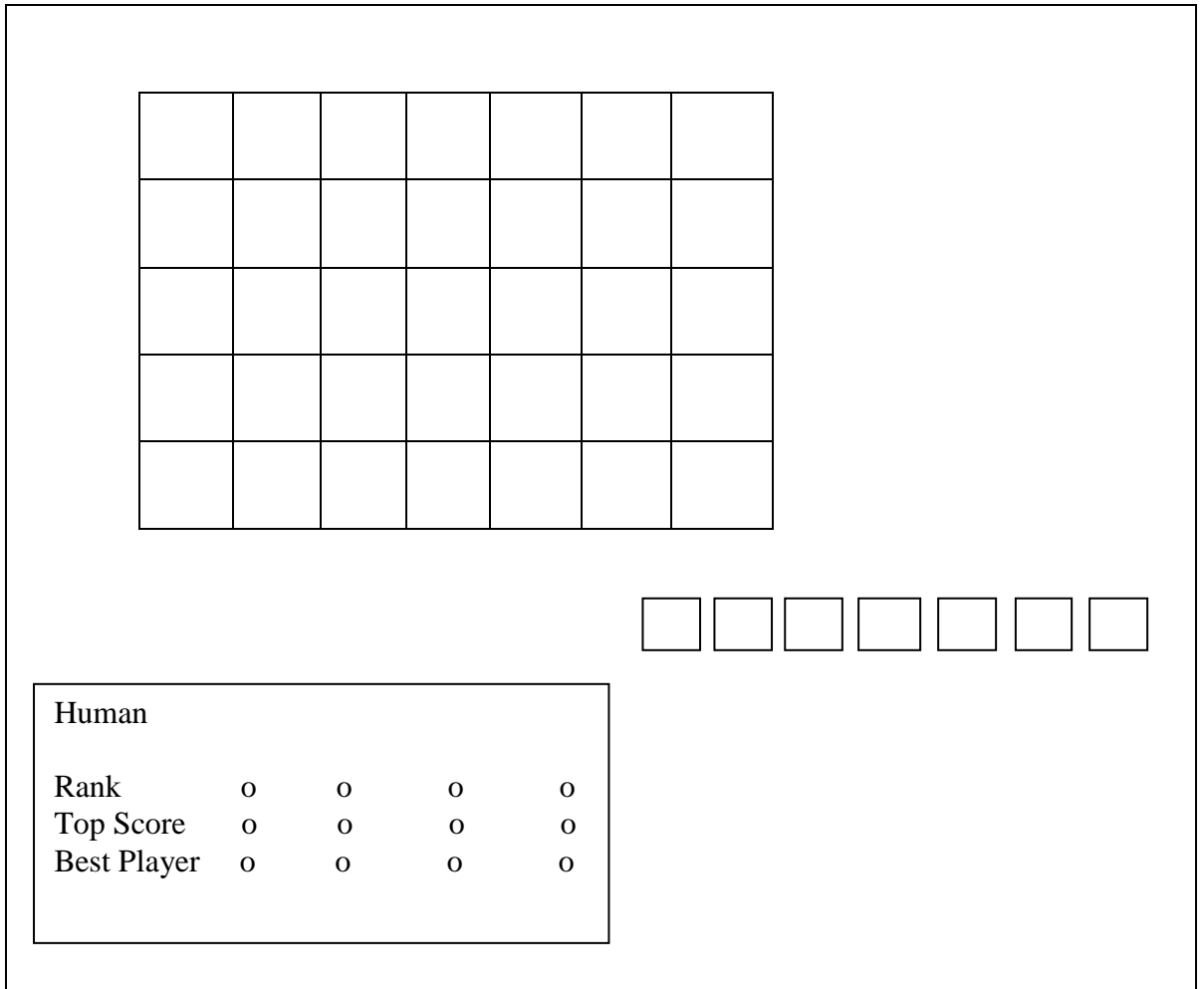


Fig 4.1, Output specification of the system

4.3 Input Specification and Design

The input design is a kits comprising of the tiles and the rack. The kit is the instrument used in playing the game. The scrabble rack is positioned on the down part of kit with tiles placed on the rack. Each letter on the rack is designed in such a way that the user can make input to the system by clicking on the letter and then click where to place it on the board.

The input design is as follows:

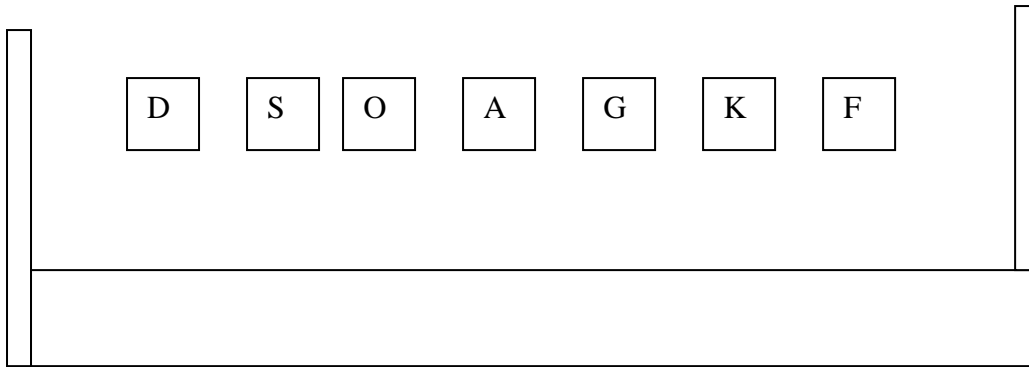


Fig 4.2, Input specification of the system

4.4 File Design

In any good database design, effort should be made to remove completely or at worst reduce redundancy. The database design in the software is achieved using Microsoft access database. Bellow is the structure of the file designed in the database.

4.4.1 DICTIONARY TABLE

FIELD	FIELD TYPE	FIELD SIZE
Word	Text	50

Table 4.1, Dictionary table

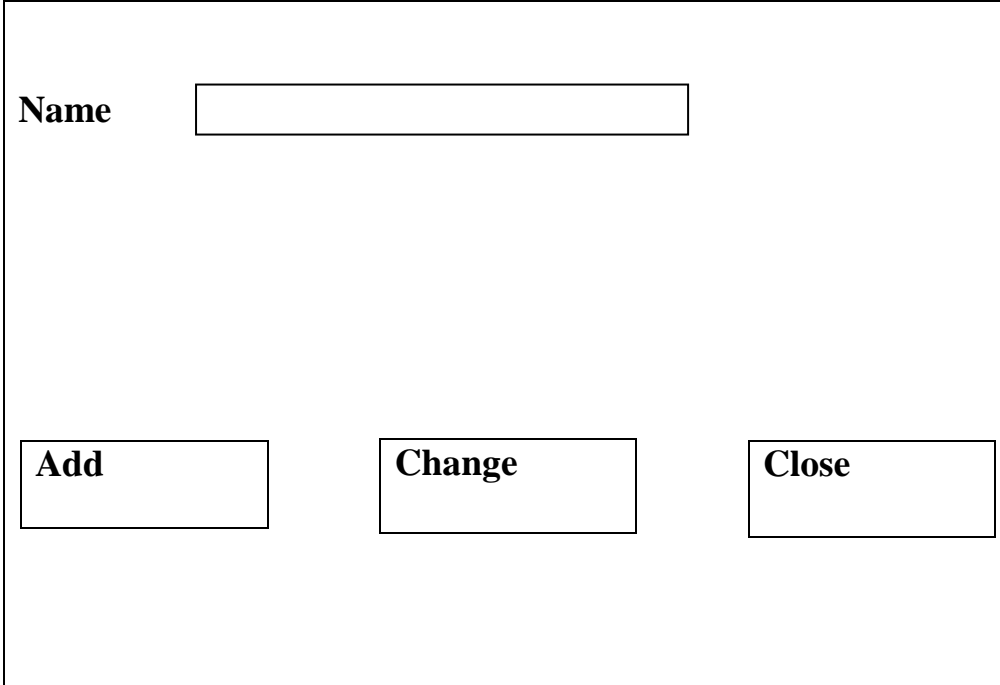
4.4.2 PLAYERS TABLE

FIELD	FIELD TYPE	FIELD SIZE
Name	Text	50
Score	Integer	2

Table 4.2, Player table

4.5 Project Modules

- Solitaire module : This is the module for one player only. This is where you select your player just for one man game.



The diagram shows a rectangular window for the Solitaire module. At the top left, the label "Name" is positioned to the left of a horizontal text input field. Below this, there are three buttons arranged horizontally: "Add" on the left, "Change" in the center, and "Close" on the right. Each button is a simple rectangle with its text centered inside.

Fig 4.3, solitaire module design

- Maven module: This is the module for two players. This is where you make your selection for two players.

The diagram illustrates the Maven module design. It consists of a rectangular frame containing the following elements:

- First Player**: A text label followed by a horizontal rectangular input field.
- Second Player**: A text label followed by a horizontal rectangular input field, positioned below the first player's input.
- Ok**: A rectangular button centered below the two input fields.

Fig 4.4, Maven module design

- Human module: This is the module for four players which is the maximum that can be had in scrabble game.

First Player	<input type="text"/>
Second Player	<input type="text"/>
Third Player	<input type="text"/>
Fourth Player	<input type="text"/>
	<input type="button" value="ok"/>

Fig 4.5, Human model design

- How to play module: This is the module that displays the steps on how to play the game.

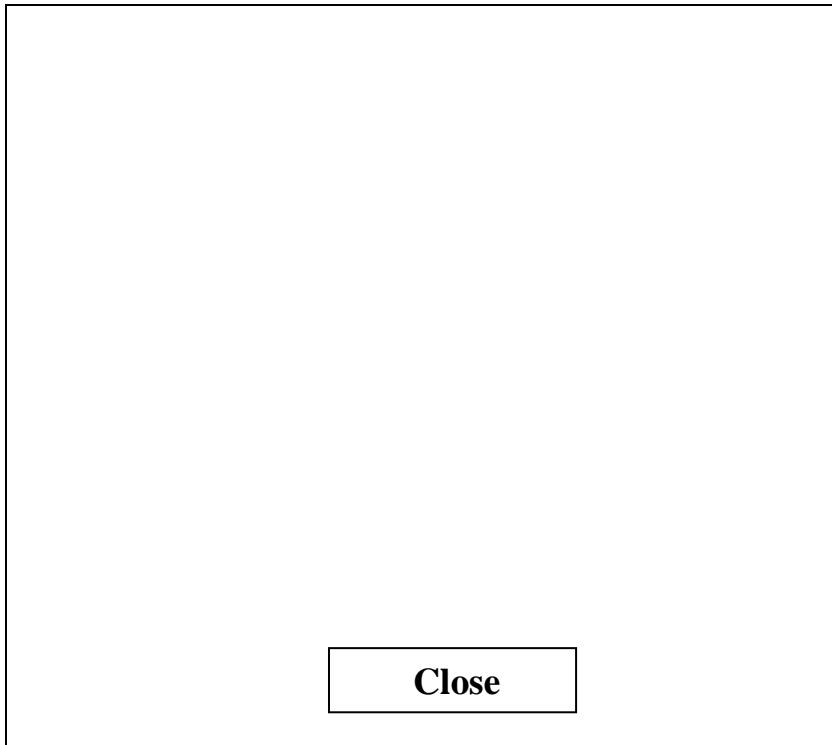


Fig 4.6, How to play module design

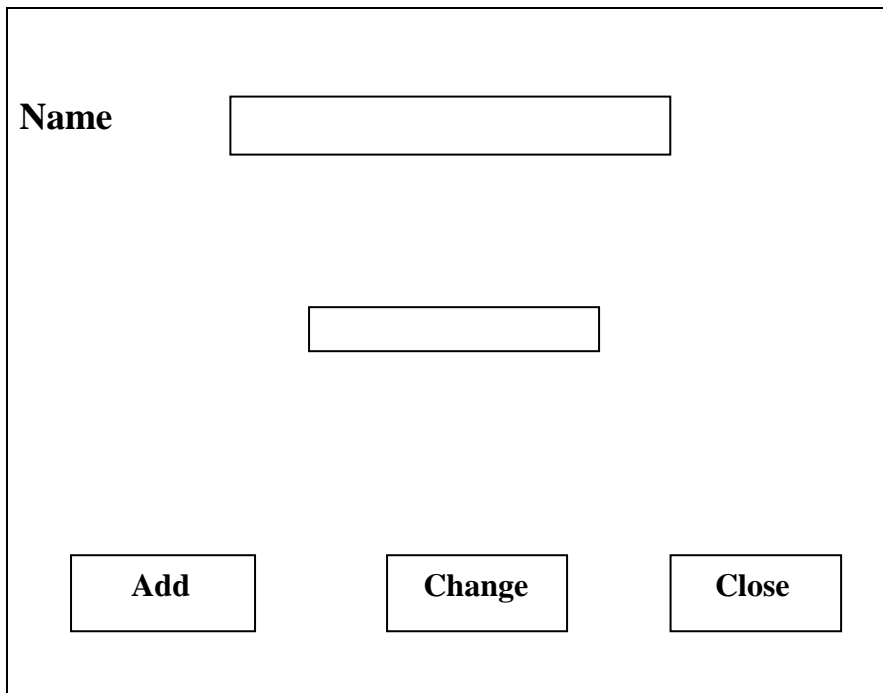
- Dictionary module : This is where you build your words or vocabulary for the game.

Word

words

Fig 4.7, Dictionary module design

- Change player module: This is where to change players when playing the game.



The diagram shows a rectangular window for the 'Change player module'. It contains the following elements:

- A label **Name** on the left side.
- A horizontal text input field to the right of the **Name** label.
- A second horizontal text input field centered below the first one.
- Three buttons at the bottom: **Add**, **Change**, and **Close**, arranged horizontally from left to right.

Fig 4.8, Change player module design

- **Game module:** This is the game environment where the game is played.

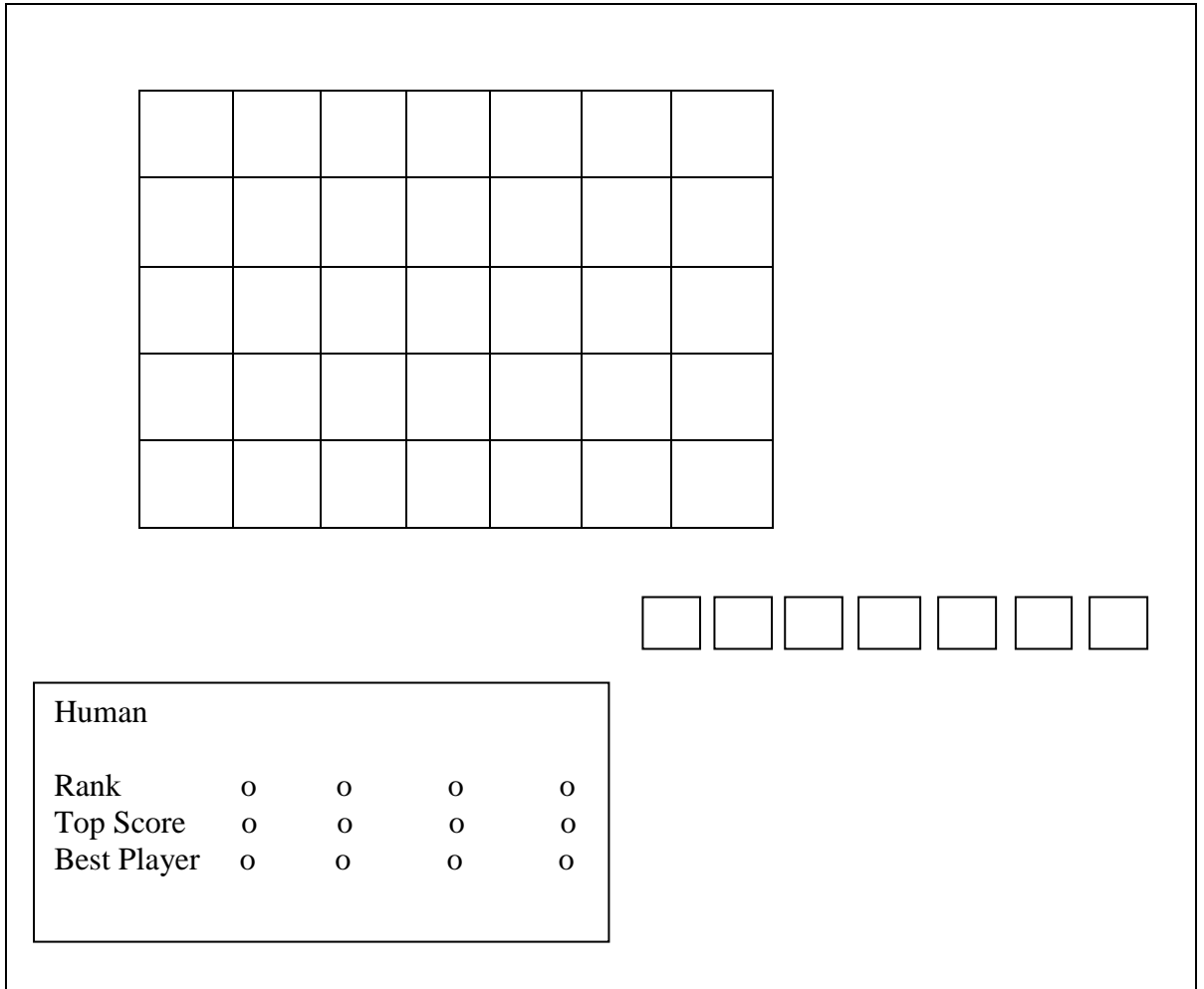


Fig 4.9, Game module design

4.6 Flowchart of operation

- **System Flowchart**

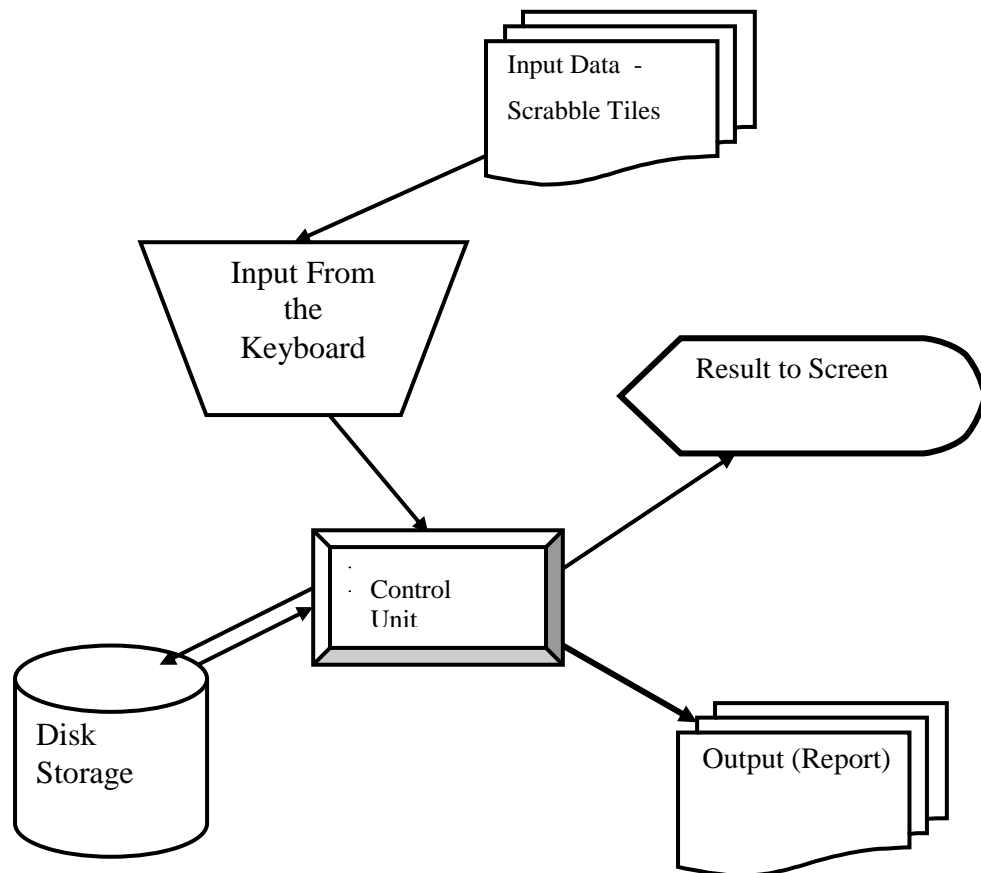


Fig 4.10, System flowchart

The scrabble tiles(letters) are entered from the keyboard and then the control unit will display them on the screen as the output. Words are also stored and retrieved to the control unit from the disk storage device.

- **Program Flowchart**

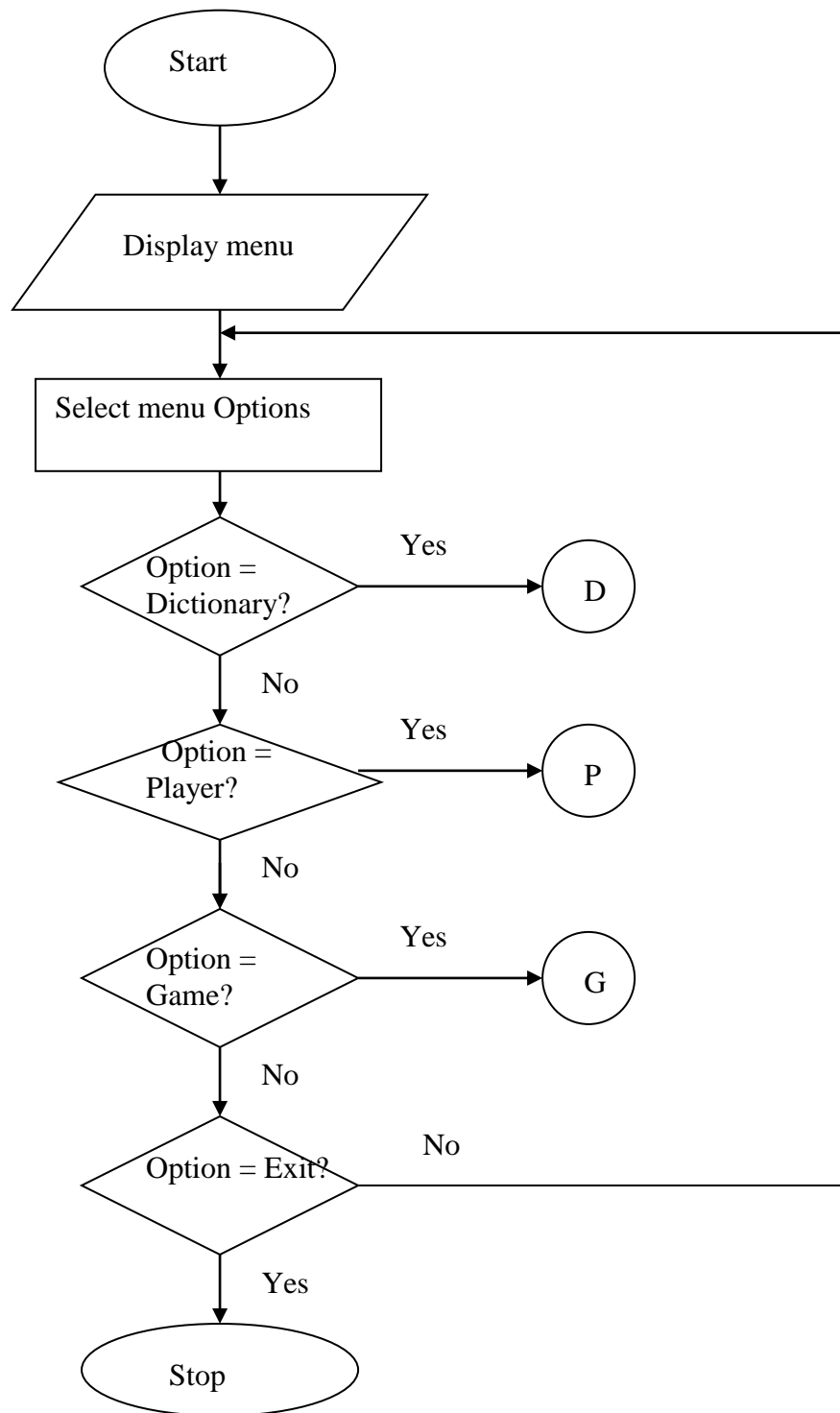


Fig 4.11, Program flowchat

When you start the game, menu will be displayed for you to select your options, whether dictionary, player, game, if Yes, you continue the game but if No, you exit.

- **Dictionary Module Flowchart**

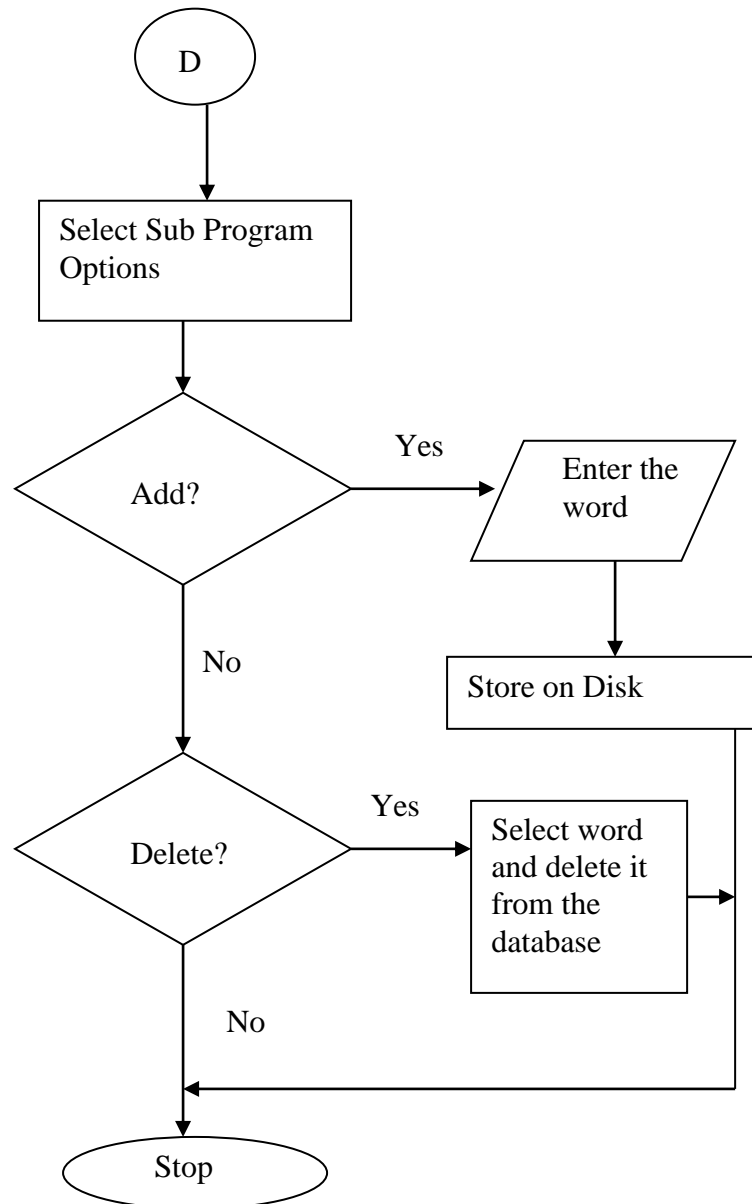


Fig 4.12, Dictionary module flowchart

Dictionary consist of add and delete options. From add option you can enter your words and store, and from delete option you select the word and delete.

- **Player Module Flowchart**

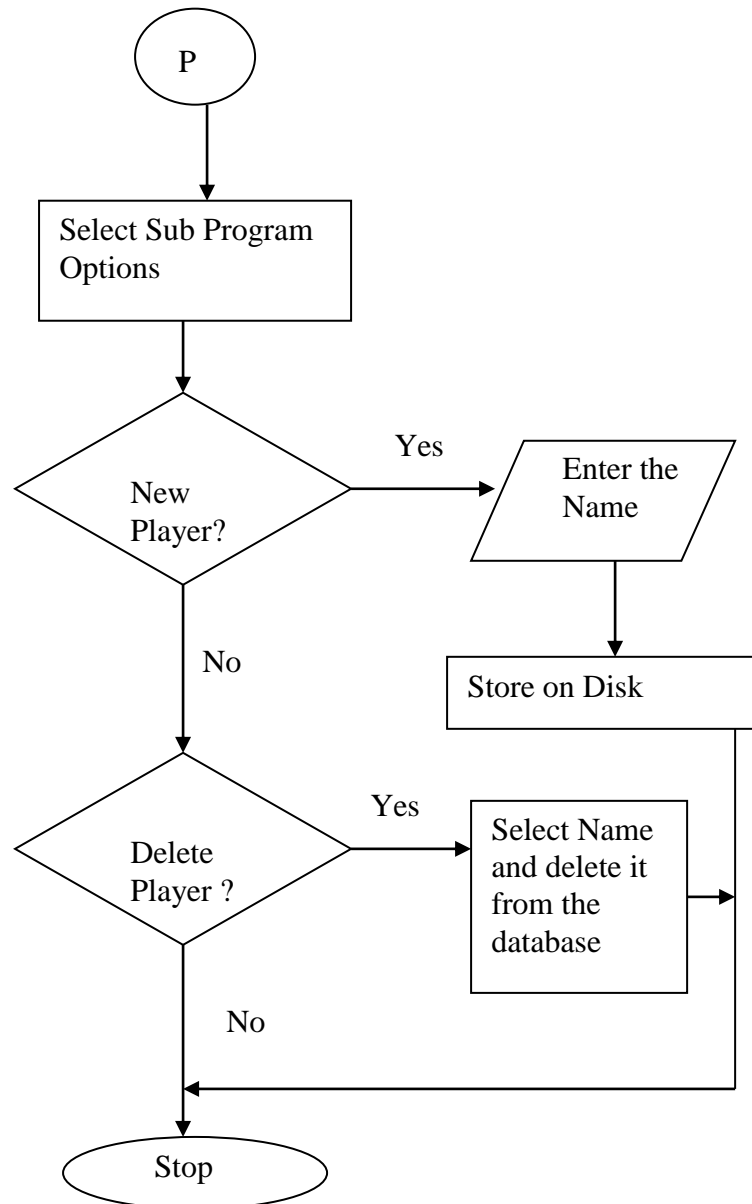


Fig 4.13, Player module flowchart

Player module consist of add and delete player options from which you can enter new players and store and also select players from the database and delete.

- **Game Module Flowchart**

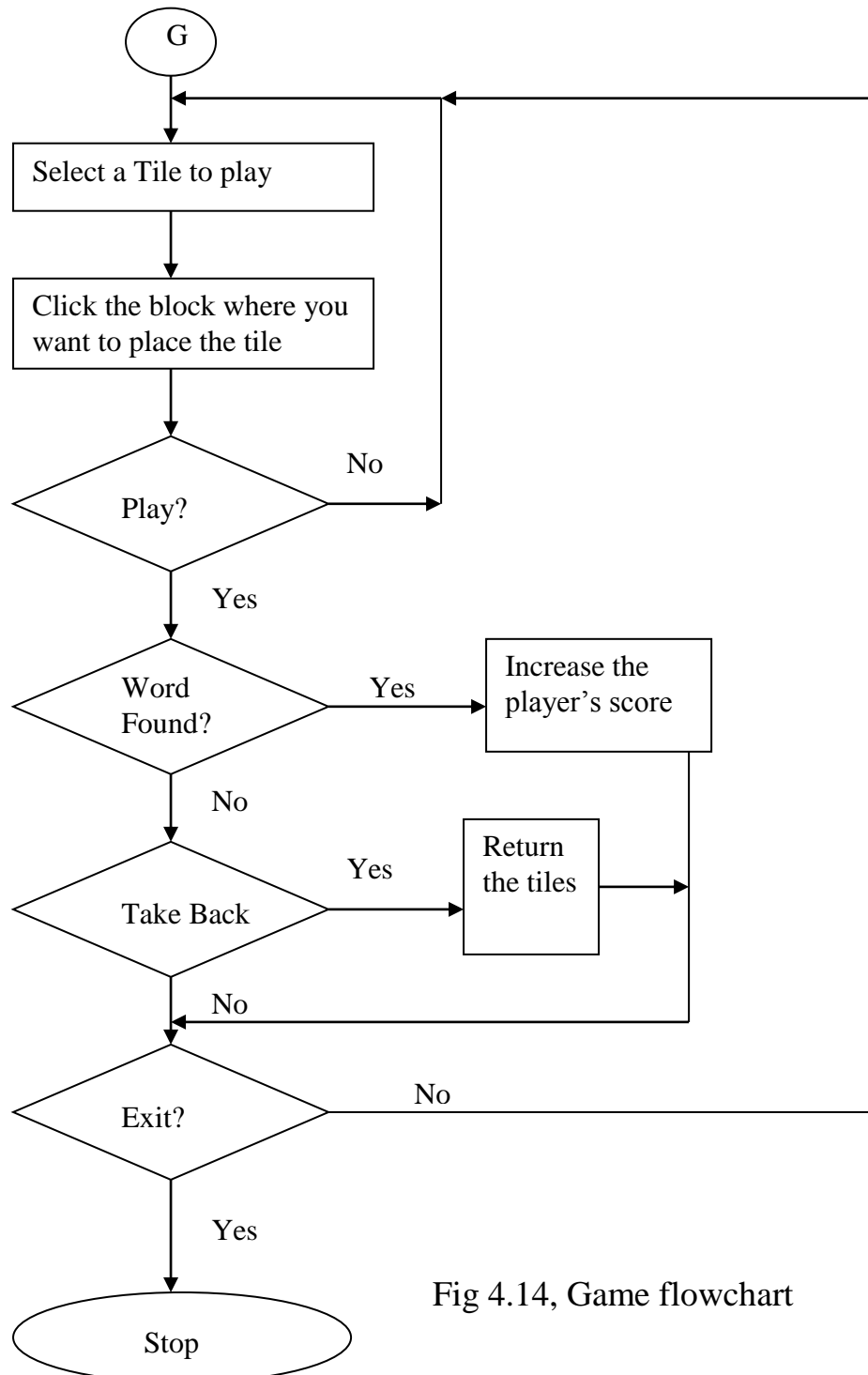


Fig 4.14, Game flowchart

As it appears on the flowchart in fig 4.14 above, if you select your tiles, you place them on the rack to form a word and then play. If the word is found in the dictionary you will be scored but if not, you can take back the tiles and form a new one if Yes, but if No you can exit the game.

CHAPTER FIVE

SYSTEM IMPLEMENTATION AND TESTING

System implementation is the process of defining the user requirements and designing a system to meet them. This is the stage of systems development in which hardware and software are acquired, developed and installed.

5.1 System Requirements

The newly developed system requires both hardware and software to run.

5.1.1 Hardware Requirement

The hardware requirement for this software is a Pentium IV computer system or higher.

5.1.2 Software Requirement

The software requirement for this system is window XP or higher version.

5.2 System module Implementation

The various modules are implemented in scrabble game,

- **Solitaire module:** This is the implementation of fig 4.3. This module is for one player only. When you choose solitaire, it means you want a one man game. From here, you can still add and change player.

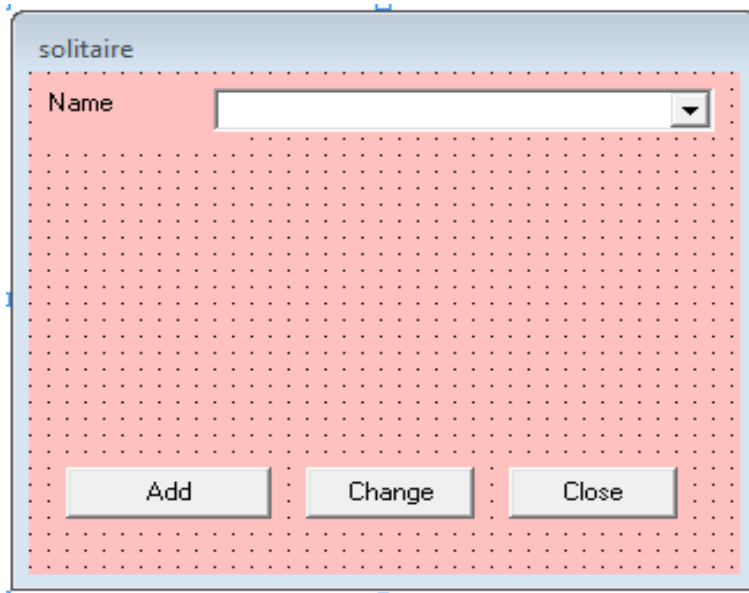


Fig 5.1, Implementation of solitaire module

- **Maven module:** This is the implementation of fig 4.4. This module is for two players. This is where you make your selection for two players. It can be called a two man game module. When you choose your first and second player, you click on OK to start.

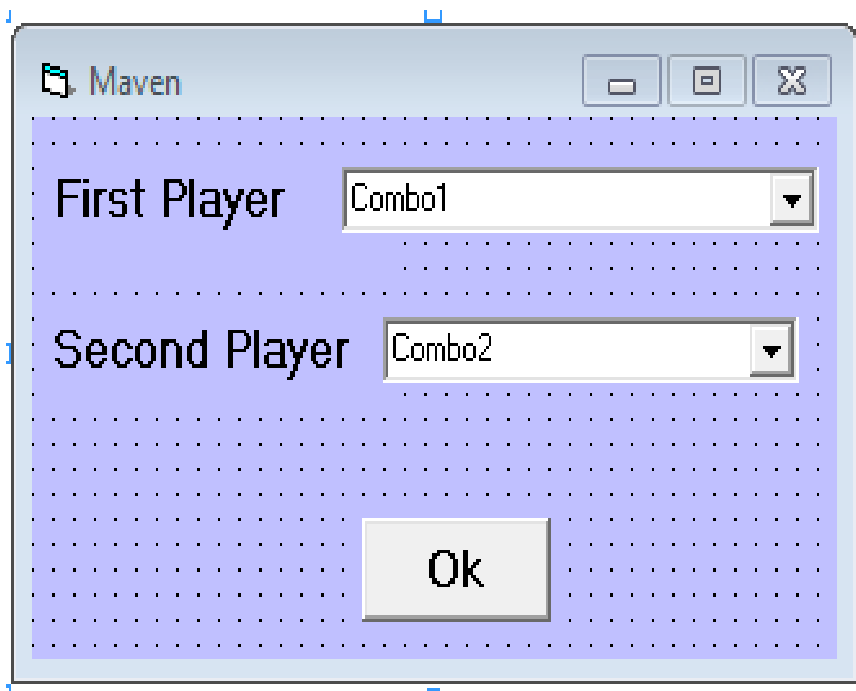


Fig 5.2, Implementation of maven module

- **Human module:** This is the implementation of fig 4.5. This module is for four players which is the maximum we can have in scrabble game. It can be a four man game module. From here, you choose your first player, second, third and fourth player and click OK to start playing.

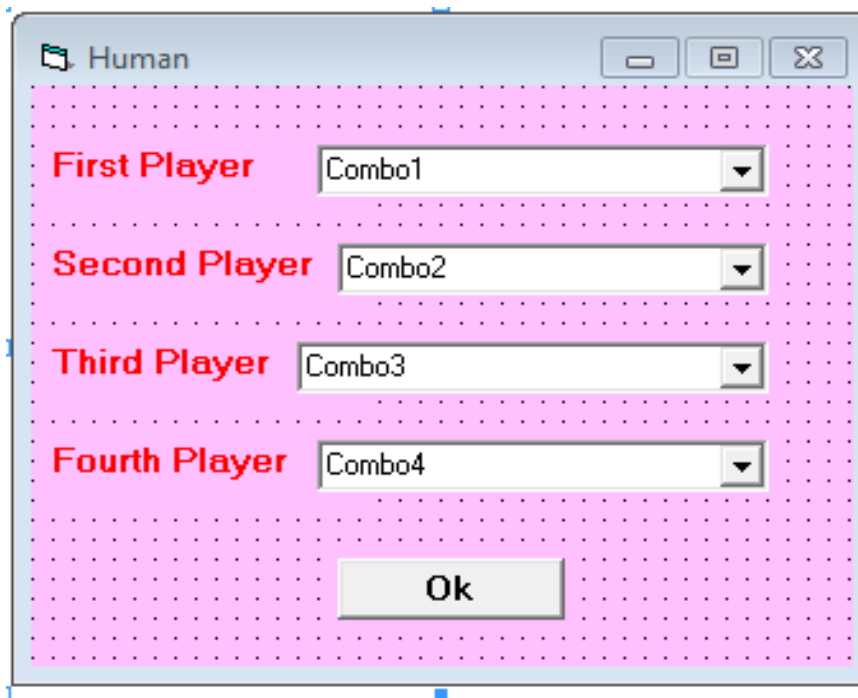


Fig 5.3, Implementation of human module

- **How to play module:** This is the implementation of fig 4.6. This is the module that displays the steps on how to play the game. By following the instructions you can be able to play this game conveniently.

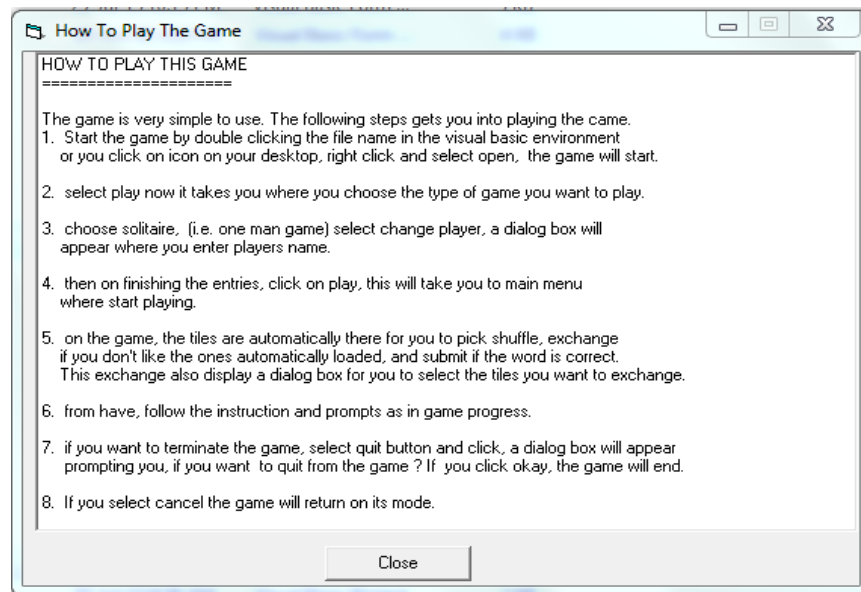


Fig 5.4, Implementation of how to play module

- **Dictionary module:** This the implementation of fig 4.7. One particularly key part of the implementation of any Scrabble game is the dictionary. Without this, there is no way for an intelligent agent to determine if a placement of tiles is valid or not. Since an agent must access the dictionary many times in determining a move, and since the dictionary is bound to be very large, performance in accessing valid words is very important.

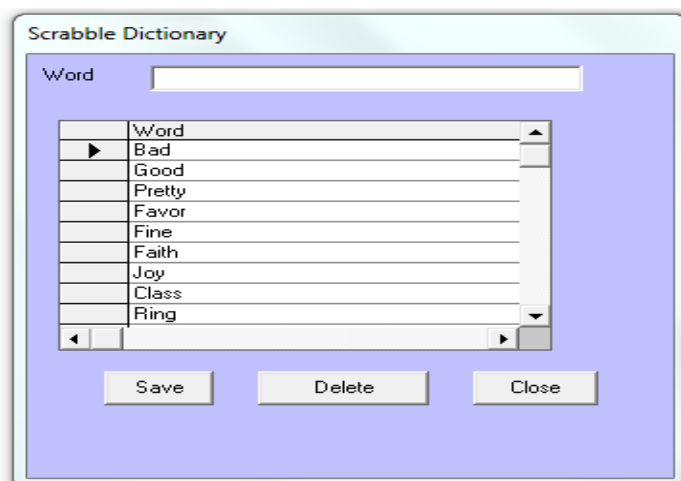


Fig 5.5, Implementation of dictionary

- **Change player module:** This is the implementation of fig 4.8. Change player module is used to change players or to add players.

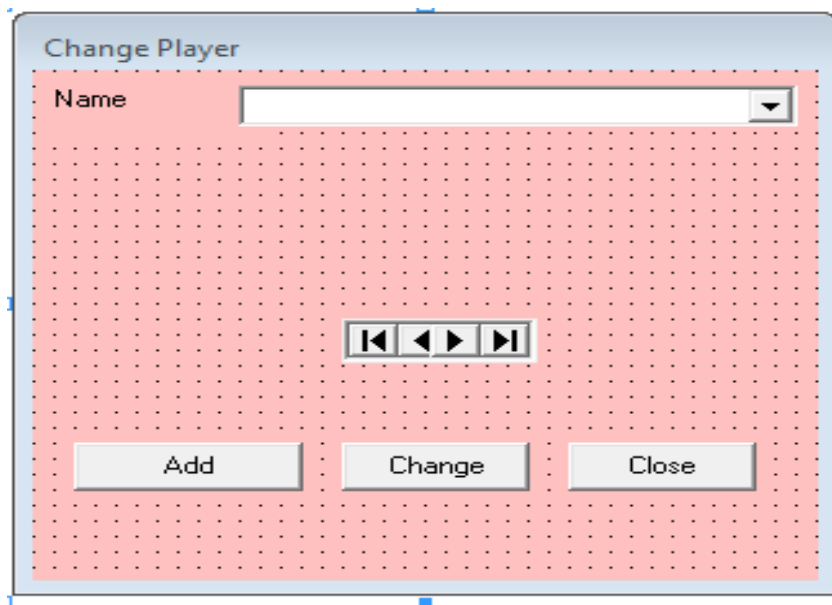


Fig 5.6, Implementation of change player module

- **Game module:** This is the implementation of fig 4.9. This is the game environment where the game is being played. The environment for each agent or player is partially observable. Players do not know what tiles are possessed by other players or which tiles still remain in the tile bag. The boards and all turns already played are fully observable on the board at all times.

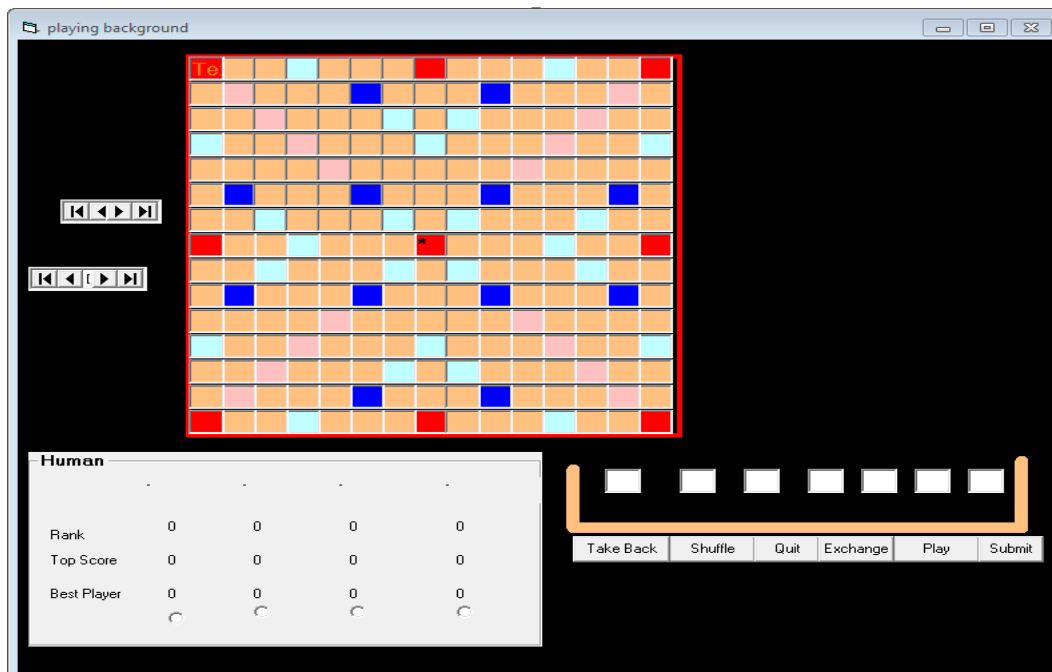


Fig 5.7, Implementation of game module

5.3 System Testing

After the integration of modules, the whole system was tested. The scrabble game was played on the system and its performance was accurate. The system testing lasted for some days and the system performance was ok.

5.3.1 Test plan

The testing is to reveal the inherent weakness in the proposed system. Real data are entered to test the efficiency and effectiveness of the system before the new system is finally implemented. Several activities must be carried out for it to work successfully. These Include:

5.3.2 Module Testing

This is the process of testing the system module by module to ascertain the functionality of each of the modules. In this testing, the following modules were tested.

- Players Module
- Game Module
- Score Module
- User Guide Module

5.3.3 Integration Test

The modules were integrated together and tested. Access to the modules was made available through the menu designed. The testing worked perfectly and the system designed is a workable one.

5.4 Change over Procedures

This is the process of changing from the manual system to computerized system. There are many methods of change over which include:

5.4.1 Direct Changeover

In this method the old system is completely replaced by the new system in one move. This may be avoidable where the two systems are substantially different, where the new system is a real time system, or when an extra staff to oversee its parallel running is unobtainable. This

method is comparatively cheap but is risky. Program corrections are difficult while the system has to remain operational. The new system should be introduced during slack periods and in large systems. It may introduce application, allowing several months between each stage to ensure all problems are cleared up before the whole system becomes operational.

5.4.2 Parallel Changeover

In this method , both the manual and computerized system are operated concurrently for sufficiently long period and their outputs compared periodically and possible discrepancies reconciled on the new system until all users are satisfied .The old system is discontinued when discrepancies are seen to have seized arising. It has the advantage of having an old system to fall back on, incase the new system fails. The disadvantage is the cost of running two systems side by side, both of which will achieve similar result.

5.4.3 Phase Changeover

Here, the changeover starts with a department or branch. The effect of the new system in the sample department or branch is observed before some other department or branch which may be more sensitive can adapt to the new system .

5.4.4 Pilot Changeover

In this case, some transactions that are very complex are operated using parallel changeover and in other remaining existing system in application, direct changeover is used.

5.5 Bill of Engineering Measurement and Evaluation (BEME)

ITEMS	AMOUNT (#)
Cost of transportation	6,000
Feeding	6,000
Software component used (Code Libries).	8,000
Software tools employed (VB 6.0, MS Access).	11,500
Testing and benchmarking	5,000
Cost of online research	5,000
Laptop	90,000
TOTAL	131,500

Table 5.1, BEME table

CHAPTER SIX

SUMMARY, CONCLUSION AND RECOMMENDATIONS

6.1 Summary

At the end of this project work, the software that can successfully simulate scrabble game was developed. The graphical interface was beautified, and the logic simulation well structured as to maintain a high level of accuracy in game scoring and declaring of the winner.

This project work also will serve as a stepping stone for people who wish to research more on this topic. The procedures for developing games are difficult, but has a lot of fun attached to it as there are so many things to benefit from playing computer games, especially the scrabble game as explained in the earlier chapters.

5.2 Conclusion

The development of computer games involves many phases. The approach used is a top-down, one concentrating on what first, then how and moving to successive levels of details.

The first phase started with a detailed study of the problems and rules for playing games. In the course of this project, many problems were discovered to have hindered the effectiveness of the existing system. These problems, information needs, and activities were documented and later used as the basis for system design, which immediately followed the first phase. The design phase was concerned primarily with the specification of the system elements in a manner that best met scrabble

game rules. During this phase, strict adherence was made on proven software engineering principles and practices. To implement this design, a computer program was then written and tested in the visual basic 6.0 programming environment. This software product is designed to run under the Microsoft Windows operating system environment.

It is hoped that effective implementation of this software product would eliminate many problems discovered during systems investigation. This will no doubt result in improved game simulation and catching of fun among players.

5.3 Recommendations

The following recommendations are made:

Playing of scrabble game should be encouraged as it helps to build students logical reasoning.

Scrabble game should be simulated and marketed so as to gain popularity among computer games.

The use of computer in playing games helps to reduce stress and should be encouraged.

REFERENCES

- Appel, A.W., Jacobsen, G.J. (1985). *The World's Fastest Scrabble Program*, *Commun. ACM*, 31(5).
- Banks, J., Carson, J.S., Nelson B.L., and Nicol D.M. (2000). *Discrete Event System Simulation*, 3rd Ed., Prentice-Hall.
- Gordon, S. A. (1993). *A Faster Scrabble Move Generation Algorithm*, *Software - Practice and experience*, Vol. 24(2).
- Ingalls, R.G. (1998). *The Value of Simulation in Modeling Supply Chains*. Proceedings of the 1998 Winter Simulation Conference.
- Ingalls, R.G., Eckersley C. (1992). *Simulation Issues in Electronics Manufacturing*. Proceedings of the 1992 Winter Simulation Conference.
- Law, A.M., Kelton W.D. (2000). *Simulation Modeling and Analysis*, 3rd ed. McGraw-Hill.
- Manning C. D., Raghavan P. (2008). *Introduction to Information Retrieval*, Cambridge University Press.
- North American SCRABBLE Players Association official tournament rules, <http://www.scrabbleplayers.org/wiki/images/a/af/Rules-20110605.pdf>, April 2012.
- Russel S., Norvig P. (2009). *Artificial Intelligence: A Modern Approach* 3rd ed. Prentice Hall.

Sheppard B. (2002). *Towards Perfect Play of Scrabble*, Maastricht.

Scrabble letter points, http://en.wikipedia.org/wiki/Scrabbleletter_distributions, April 2012

World English-Language Scrabble Players' Association (WESPA)
rules, <http://www.wespa.org/rules/RulesV2nov11.pdf>, April 2012

APPENDIX A

Source code

For solitaire

```
Private Sub cmdadd_Click()
```

```
Dataplayer.Recordset.AddNew
```

```
Dataplayer.Recordset.Fields("Name") = Cmbplayer.Text
```

```
Dataplayer.Recordset.Fields("score") = 0
```

```
Dataplayer.Recordset.Fields("view") = "Yes"
```

```
Dataplayer.Recordset.Update
```

```
MsgBox "Operation Completed", vbOKOnly, "Success"
```

```
End Sub
```

```
Private Sub cmdchange_Click()
```

```
frmchange.Hide
```

```
frmgame.Show
```

```
End Sub
```

```
Private Sub cmdclose_Click()
```

```
frmchange.Hide
```

```
frmgame.Show
```

```
End Sub
```

For maven

```
Private Sub cmdok_Click()
```

```
frmmav.Hide
```

```
frmgame.Show
```

```
End Sub
```

For human

```
Private Sub cmdok_Click()
```

```
frmhuman.Hide
```

```
frmgame.Show
```

```
End Sub
```

For Dictionary

```
Private Sub Command1_Click()
```

```
End Sub
```

```
Private Sub cmdclose_Click()
```

```
frmdictionary.Hide
```

```
frmgame.Show
```

```
End Sub
```

```
Private Sub cmddelete_Click()
```

```
Data1.Recordset.Delete
```

```
End Sub
```

```
Private Sub cmdsave_Click()
```

```
Data1.Recordset.AddNew
```

```
Data1.Recordset.Fields("word") = txtword.Text
```

```
Data1.Recordset.Update
```

```
txtword.Text = ""
```

```
Data1.Refresh
```

```
DBGrid1.Refresh
```

```
End Sub
```

For presenter

```
Private Sub Timer1_Timer()
```

```
Me.Hide
```

```
Me.Timer1.Enabled = False
```

```
frmwelcomepage.Show
```

```
End Sub
```

For welcome page

```
Private Sub cmdplay_Click()
```

frmgame.Show

Me.Hide

End Sub

For selection page

Option Base 1

Option Explicit

Public i As Integer

Private Sub cmddictionary_Click()

frmdictionary.Show

Me.Hide

End Sub

Private Sub cmdexit_Click()

MsgBox "Good Bye ", vbCritical, "Exit"

End

End Sub

Private Sub cmdhelp_Click()

frmhow.List1.Clear

frmhow.List1.AddItem "HOW TO PLAY THIS GAME"

frmhow.List1.AddItem "=====

frmhow.List1.AddItem ""

frmhow.List1.AddItem "The game is very simple to use. The following steps gets you into playing the came."

frmhow.List1.AddItem "1. Start the game by double clicking the file name in the visual basic environment "

frmhow.List1.AddItem " or you click on icon on your desktop, right click and select open, the game will start."

frmhow.List1.AddItem ""

frmhow.List1.AddItem "2. select play now it takes you where you choose the type of game you want to play."

frmhow.List1.AddItem ""

frmhow.List1.AddItem "3. choose solitaire, (i.e. one man game) select change player, a dialog box will "

frmhow.List1.AddItem " appear where you enter players name."

frmhow.List1.AddItem ""

frmhow.List1.AddItem "4. then on finishing the entries, click on play, this will take you to main menu "

frmhow.List1.AddItem " where start playing."

frmhow.List1.AddItem ""

frmhow.List1.AddItem "5. on the game, the tiles are automatically there for you to pick shuffle, exchange"

```
frmhow.List1.AddItem " if you don't like the ones automatically loaded,  
and submit if the word is correct."
```

```
frmhow.List1.AddItem " This exchange also display a dialog box for  
you to select the tiles you want to exchange."
```

```
frmhow.List1.AddItem ""
```

```
frmhow.List1.AddItem "6. from have, follow the instruction and prompts  
as in game progress."
```

```
frmhow.List1.AddItem ""
```

```
frmhow.List1.AddItem "7. if you want to terminate the game, select quit  
button and click, a dialog box will appear"
```

```
frmhow.List1.AddItem " prompting you, if you want to quit from the  
game ? If you click okay, the game will end."
```

```
frmhow.List1.AddItem ""
```

```
frmhow.List1.AddItem "8. If you select cancel the game will return on  
its mode."
```

```
frmhow.Show
```

```
frmgame.Hide
```

```
End Sub
```

```
Private Sub Command3_Click()
```

```
On Error Resume Next
```

```
frmchange.Cmbplayer.Clear
```



```
frmchange.Dataplayer.Recordset.MoveFirst
Do Until frmchange.Dataplayer.Recordset.EOF
frmchange.Cmbplayer.AddItem
frmchange.Dataplayer.Recordset.Fields("Name")
frmchange.Dataplayer.Recordset.MoveNext
Loop
frmchange.Show
Me.Hide
End Sub
```

```
Private Sub Command5_Click()
```

```
Dim i As Integer
```

```
Dim wel As String
```

```
Dim t As TextBox
```

```
frmgame.Data3.Refresh
```

```
frmgame.Data2.Refresh
```

```
frmscrabble.Data3.Refresh
```

```
frmscrabble.Data1.Refresh
```

```
wel = " "
```

```
If (Option1.Value = True) And (frmchange.Cmbplayer.Text =
"Combo1") Then
```

```
MsgBox "Please Select a player", vbCritical, "Error"
```

```
Else
```

```
frmscrabble.Show
```

```
For i = 0 To 224
```

```
frmscrabble.txtscrabble(i).Text = ""
```

```
Next i
```

```
frmscrabble.txtscrabble(147).Text = "*"
```

```
frmscrabble.txtscrabble(147).SetFocus
```

```
If Option1.Value = True Then
```

```
frmscrabble.lblplayer.Caption = frmchange.Cmbplayer.Text
```

```
wel = wel + frmchange.Cmbplayer.Text
```

```
End If
```

```
If Option2.Value = True Then
```

```
frmscrabble.Frame1.Visible = True
```

```
frmscrabble.Label1.Caption = frmnav.Combo1.Text
```

```
frmscrabble.Label2.Caption = frmnav.Combo2.Text
```

```
wel = wel + frmnav.Combo1.Text + ", " + frmnav.Combo2.Text
```

```
End If
```

```
If Option3.Value = True Then
```

```
frmscrabble.Frame2.Visible = True
```

```

frmscrabble.Label18.Caption = frmhuman.Combo1.Text
frmscrabble.Label19.Caption = frmhuman.Combo2.Text
frmscrabble.Label20.Caption = frmhuman.Combo3.Text
frmscrabble.Label21.Caption = frmhuman.Combo4.Text
wel = wel + frmhuman.Combo1.Text + ", " + frmhuman.Combo2.Text +
", " + frmhuman.Combo3.Text + ", " + frmhuman.Combo4.Text
End If

End If

wel = "Hi " + wel + ", welcome to Scrabble Game"
MsgBox wel, vbInformation, "Welcome"

'frmchange.Dataplayer.Recordset.MoveFirst
frmscrabble.lblscore.Caption = 0
'frmgame.lblrank.Caption =
frmchange.Dataplayer.Recordset.Fields("score")
'Exit Do
'Else
'frmchange.Dataplayer.Recordset.MoveNext
'End If
'Loop
frmgame.Hide

```

End Sub

Private Sub Option1_Click()

On Error Resume Next

frmchange.Cmbplayer.Clear

frmchange.Dataplayer.Recordset.MoveFirst

Do Until frmchange.Dataplayer.Recordset.EOF

frmchange.Cmbplayer.AddItem

frmchange.Dataplayer.Recordset.Fields("Name")

frmchange.Dataplayer.Recordset.MoveNext

Loop

frmchange.Show

Me.Hide

End Sub

Private Sub Option2_Click()

On Error Resume Next

frmnav.Combo1.Clear

frmnav.Combo2.Clear

frmchange.Dataplayer.Recordset.MoveFirst

Do Until frmchange.Dataplayer.Recordset.EOF

```
frmnav.Combo1.AddItem  
  
frmchange.Dataplayer.Recordset.Fields("Name")  
  
frmnav.Combo2.AddItem  
  
frmchange.Dataplayer.Recordset.Fields("Name")  
  
frmchange.Dataplayer.Recordset.MoveNext  
  
Loop  
  
frmnav.Show  
  
frmgame.Hide  
  
End Sub
```

```
Private Sub Option3_Click()  
  
On Error Resume Next  
  
frmhuman.Combo1.Clear  
  
frmhuman.Combo2.Clear  
  
frmhuman.Combo3.Clear  
  
frmhuman.Combo4.Clear  
  
frmchange.Dataplayer.Recordset.MoveFirst  
  
Do Until frmchange.Dataplayer.Recordset.EOF  
  
frmhuman.Combo1.AddItem  
  
frmchange.Dataplayer.Recordset.Fields("Name")  
  
frmhuman.Combo2.AddItem  
  
frmchange.Dataplayer.Recordset.Fields("Name")
```

```
frmhuman.Combo3.AddItem  
  
frmchange.Dataplayer.Recordset.Fields("Name")  
  
frmhuman.Combo4.AddItem  
  
frmchange.Dataplayer.Recordset.Fields("Name")  
  
frmchange.Dataplayer.Recordset.MoveNext  
  
Loop  
  
frmhuman.Show  
  
frmgame.Hide  
  
End Sub
```

APPENDIX B

User Guide

When you lunch the scrabble game installed in your system, the presenter page will appear first which shows the topic of the project and the name of the presenter, after some seconds the welcome page follows. From the welcome page you click on the “Play Now” button which takes you to the setting page where you do all the necessary settings and also choose how many player game you want to play, whether one man game (solitaire), two man game (Mavem) or four man game (Huma). You can click on the dictionary button to view the word you have already stored in the

dictionary. Click on “How to Play” button to view the instruction on how to play the game. After all these settings, you now click on the play button to start the game. This project is designed in such a way that every game starts from the center of the board that has the star symbol in it. After each word formed on the game board, you click on play button to score you. At the end of the game you click on the submit button so that your score will be shown on the setting page, the click on the quit button to exit the game.