KOLEJ UNIVERSITI KEJURUTERAAN DAN TEKNOLOGI MALAYSIA

BORANG PENGESAHAN STATUS TESIS

JUDUL **CHESS TOURNAMENT MANAGEMENT SYSTEM (PAIRING MODULE)**

SESI PENGAJIAN: **2004/2005**

Saya **YEOW CHEE YOU**
(HURUF BESAR)

mengaku membenarkan tesis (PSM/Sarjana/Doktor Falsafah)* ini disimpan di Perpustakaan Kolej Universiti Kejuruteraan dan Teknologi Malaysia dengan syarat-syarat kegunaan seperti berikut:

1. Tesis adalah hakmilik Kolej Universiti Kejuruteraan dan Teknologi Malaysia.
2. Perpustakaan Kolej Universiti Kejuruteraan dan Teknologi Malaysia dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. **Sila tandakan (✓)

☐	SULIT	(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)
☐	TERHAD	(Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)
☑	TIDAK TERHAD	

Disahkan oleh

(TANDATANGAN PENULIS) (TANDATANGAN PENYELIA)

Alamat Tetap: **NO 1, JLN TOMAN 6,**
TMN SERI PUTRA,
42700 BANTING, SEL

SUPERVISOR
Wan Muhammad Syahrir

Tarikh: 22nd March 2005 Tarikh: 22nd March 2005

CATATAN:
* Potong yang tidak berkenaan.
** Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali sebab dan tempoh tesis ini perlu dikelaskan sebagai SULIT atau TERHAD.
◆ Tesis dimaksudkan sebagai tesis bagi Ijazah Doktor Falsafah dan Sarjana secara penyelidikan, atau disertai bagi pengajian secara kerja kursus dan penyelidikan, atau Laporan Projek Sarjans Muda (PSM).

"I hereby acknowledge that I had read this technical writing and in my opinion this technical writing is sufficient in terms of scope and quality for the purpose of the granting of Bachelor of Computer Technology (Software Engineering)."

Signature : .....

Name of Supervisor : Wan Muhammad Syahrir

Date : 22nd March 2005

CHESS TOURNAMENT MANAGEMENT SYSTEM (PAIRING MODULE)

YEOW CHEE YOU

A thesis submitted in fulfilment of the
requirements for the award of the degree of
Bachelor of Computer Technology (Software Engineering)

Faculty of System Computer & Software Engineering
University College of Engineering & Technology Malaysia

MARCH, 2005

I declare that this thesis entitled "Chess Tournament Management System (Pairing Module)" is the result of my own research except as cited in the references. The thesis has not been accepted for any degree and is not currently submitted in candidature of any other degree.

Signature	:	.....
Name	:	YEOW CHEE YOU
Date	:	8/4/2005

ACKNOWLEDGEMENT

First and foremost, I wish to express my sincere gratitude to my main thesis supervisor, En. Wan Muhammad Syahrir Bin Wan Hussin, for his inspiration, guidance and critics that he has given me during my thesis research. The idea sharing, encouragement, and team working provided by my team members, William Khor Kieng Ann and Lim Jiin Kang helps a lot in solving some of the problems that were encountered in the chess pairing method are greatly appreciated. Without their continued support and interest, the thesis would not have been the same as presented here.

Thanks also to all the lecturers from the Faculty of Computer Systems & Software Engineering and staffs from ICT team that have taken part, contributed to and supported this research. Finally and most specially thanks to my family and friends for supporting me all this time.

ABSTRACT

Chess is one of the student's favourite sports in KUKTEM. All Chess tournaments in KUKTEM are organized by chess club. Unfortunately, tournaments cannot run smoothly due to problems that occurred during the tournaments' management process. Lateness of pairing and unfair colour allocation was among the problems that should be overcome by Chess Club generally. Involvement of too many players in a particular tournament causes lateness. The organizer has to waste a lot of time in chess pairing. Besides that, problems of unfair colour allocation lead to argument among players. All players prefer white colour rather than black colour because they got better chance to win if they are white side. Furthermore , mistake always occurred when tournament comes to heavy task like chess pairing and colour allocation.This cause players complaint about the unfair treatments in the tournament. So for that, in order to overcome the problem, Chess Pairing module needs to be developed. The development of chess pairing module of Chess Tournament Management System is suggested for handling the player pairing and colour allocation in every match. A Software Development Life Cycle (SDLC) is use as a method in the development of this system.Chess Pairing module will use Gale Shapley method as a based idea to develop an algorithm to do the entire pairing and colour allocation task.

ABSTRAK

Catur merupakan salah satu sukan kegemaran pelajar KUKTEM. Semua pertandingan catur di KUKTEM adalah dianjurkan oleh kelab catur.Namun Demikian , masih terdapat masalah dalam pengendalian pertandingan sehingga pertandingan tidak dapat berjalan dengan lancar. Masalah seperti penentuan pasangan perlawanan yang lembap dan pemberian warna yang tidak adil merupakan masalah yang perlu diatasi oleh kelab catur. Kelewatan selalu berlaku apabila bilangan pemain dalam pertandingan terlalu ramai. Banyak masa diambil oleh penganjur untuk melakukan kerja penentuan pasangan perlawanan untuk semua perlawanan. Dalam masa yang sama, pemberian warna yang tidak adil juga sentiasa berlaku kerana pemberian warna sukar ditentukan dan selalu menimbulkan isu ketidakpuasan di kalangan pemain. Semua pemain lebih gemar main catur dengan warna putih daripada warna hitam kerana peluang untuk menang adalah lebih tinggi. Selain itu, kesilapan juga selalu berlaku dalam kerja-kerja berat seperti penentuan pasangan perlawanan dan pemberian warna catur. Ini menyebabkan ramai pemain mengadu tentang ketidakadilan dalam pertandingan. Maka dengan itu dicadangkan agar modul penentuan pasangan perlawanan diwujudkan bagi menyelesaikan masalah tersebut. Pembangunan system ini adalah untuk menyenangkan kerja penentuan pasangan dan warna dalam setiap perlawanan. System ini dilaksanakan dengan menggunakan kaedah *Software Development Life Cycle (SDLC)*. Dalam module ini, kaedah Gale Shapley akan digunakan untuk membangunkan algorithma yang dapat melakukan kerja penentuan pasangan dan warna.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CTMS	-	Chess Tournament Management System
KUKTEM	-	Kolej Universiti Kejuruteraan dan Teknologi Malaysia
SQL	-	Structured Query Language
IT	-	Information Technology
SDLC	-	Software Development Life Cycle
SMP	-	Symmetric Processing

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

Chess tournament management is heavy task that needed to perform in whatever chess tournament in order to perform fair treatments for all players. Organizer need to do the pairing as well as the scoring task for the chess tournament that took a lot of time. The pairing task can be defined as a task that determined which player is supposed to meet which player in the tournament. Colour allocation of chess tournament is also a crucial part in pairing task that determines whether player play in black or white side of chess board.

In fact of that, Chess Tournament Management System (CTMS) is stand –alone system that reduces the time spent on administrative tasks by simplifying all phases of chess tournament management. This system will take KUKTEM as case study. Basically, this system is designed to be divided into 3 modules which are:

a) Pairing Module
b) Scoring Module
c) Scheduling Module

Pairing module of Chess Tournament Management system is a module that designed to help handle the pairing task for all chess tournaments and provide a fair treatment for players.

1.1 Problem statement

In many chess tournament, the number of players is much larger than number of rounds to be played. In such tournament, a lot of time needed to complete the whole pairing process. There are a lot of arguments and complaints in those chess tournaments. Players feel like not treated fairly if international rules of chess tournaments weren't followed by the tournament especially in the first round pairing method. Besides that, when tournament comes to heavy and confusing task like take back round, pairing or colour allocation, mistakes always made by organizer.

1.2 Objectives

The objectives of this project are:

a) Develop an algorithm that can be used to determine the pairs in pairing system.
b) Allocate suitable colour for each player in every chess board by providing computerized colour allocation algorithm.

1.3 Scopes

The scopes of this project are:

a) Develop a module that consist new algorithm to generate pairing task for both knock out system and Swiss system.
b) Using Visual Basic.net as programming language and SQL server as the database.
c) Data limitation is within 50 players in a tournament.

CHAPTER 2

LITERATURE REVIEW

2.1 Chess Pairing System

If in a chess tournament the number of players is a lot bigger than the number of rounds to be played, two systems of chess pairing will be used in the chess tournament. They are Swiss Pairing [1] and Knock out Pairing [1].

2.1.1 Swiss Pairing system

For Swiss Pairing system, players are numbered from 1 to n according to their ratings so that the player with the highest rating gets number 1 and the player with the lowest rating gets number n. In the first round the pairs are (1, n/2), (2, n/2 + 1),..., (n/2 - 1, n). The winner of a game scores one point, the loser scores nil.

In the second round and in all rounds to come players with equal scores should play against each others. This is not always possible. An obvious reason is that there can be score groups (i. e. groups of players with equal number of points) with odd number of players. Second, the players in a score group may have already played against each others.

Third, in addition to scores we must also take care of the colours (playing with white or black pieces). As far as possible, at the end of each even round, all players should have had an equal number of whites and blacks. The colour history tells the colours with which the player has played in the previous rounds. An alternating colour history (e.g. WBWBW) is the optimal one. There is a trade-off between the demands concerning scores and colours: is it better to allow a pair with non-equal scores if it equalizes the colour histories or should we stick with equal scores although it might mean repetition of colours.

Pairing must be done score group by score group starting at the topmost group and continuing just above the middle group (the one containing the median player), then going to the lowest group and continuing upwards. Finally we handle the middle group with all the problematic cases pushed forward when the other groups were handled.

2.1.2 Knock Out Pairing System

This system is used to rapidly find the best player in a large pool of participants. This is achieved by eliminating losers of each match from the tournament. Just like Swiss System tournaments, the top player in the upper half of the field is then paired against the top player in the lower half of the field, and so on. The top-ranked player get white colour in the first round, and then colours alternate down the halves. Player who win in the match will qualified to next round and paired with next player who win his match. This flow will be continued until the winner born.

2.2 Traditional Chess Tournament Management

Currently in KUKTEM, All chess tournaments were organized by chess club. To organize a fair and no argument chess tournament, the organizer has to be a very

knowledgeable person that expert in all chess rules and tournament rules. He needs to take care of all pairing task and scoring task as well as judging task in that particular tournament so that no argument will be occurred in that tournament. Before the tournament start, organizer has to set how many round will be played. All the participants' name must first be collected. Then organizer and his members will do the pairing for first round. All players will get their board or seat. Each board will have 2 players. Each player's colour (Black or White) will be allocated by organizer and his members. Players will start play on allocated colour and then the first round complete. All result will be collected and the score will be calculated. A standing table needs to be generated to make sure all participants clear with their situation. This took time about 30-45 minutes for the organizer to produce the standing. Based on the score, pairing for next round will be generated. In chess, there are always an occasion where 2 or 3 players have same scores, thus there are a lot of international rules that used to do the pairing. After all round finished, player with the highest rank in standing table will win the tournament.

2.3 Computerized Chess Pairing System

Basically, there are very few approaches that can be implemented to do the computerized Chess Pairing. Approach that used in this module is an approach that based on algorithm named Gale Shapley. This Algorithm was developed by Gale Shapley at 16, February1975

2.3.1 Gale Shapley Algorithm

The Gale-Shapley algorithm [2] finds a stable matching in time O (n^2), where n is the common number of men and women. For each instance there exists at least one

stable matching and the maximum number of stable matching grows exponentially when n grows.

Based on Gale- Shapley algorithm that showed in figure 2.1, in the first round each man proposes to the woman whom he most prefers, even if someone else has already proposed to her. Then, from the proposals that she receives, each woman tentatively accepts the proposal from (becomes engaged to) the proposer whom she prefers the most; she rejects all the other proposals. A woman who does not receive any proposals waits for the next round.

In each subsequent round man who are currently engaged do nothing. Each man who is not engaged makes a new proposal, to the woman highest in his preference ranking who has not already rejected him, whether or not she is already engaged. In the women's part of the round, a woman accepts the proposal from the man highest in her ranking, rejecting all others and (if necessary) breaking her current engagement to become engaged to a man higher in her ranking. A woman who does not receive any proposals in this round waits for the next round.

If an arbitrary man prefers another woman to the one to whom he is matched, then he must have proposed to that other woman in some round. She must have rejected him because she preferred someone else. Hence, the man and woman in question cannot be a blocking pair. But since the man was arbitrary, and the woman was any woman whom he preferred over the one to whom he was matched, this have shown that there are no blocking pairs in the matching.

No man's proposal is affected in any way by what another man does. Likewise, no woman's choice is affected by what other women do. So, in each round it doesn't matter in what order the proposals are made or in what order the women make their choices.

When the algorithm terminates, all the men are engaged. Since each man is engaged to exactly one woman, and there are exactly as many women as men, we have a matching.

```
for each unengaged man
   send proposal to highest-ranked woman not yet proposed to
endfor
for each woman
   get engaged to most preferred man who has proposed
endfor
while there is man who is not engaged
  for each man, m
    if m is not yet engaged then   m springs into action
      w  highest on m's list to whom m has not yet proposed
      add w to m's list of women proposed to   m proposes to w
  endfor
  for each woman w    engaged or not
    if no proposers then  do nothing
    else
      m      highest among proposers to w   identify best suitor
      if w is not engaged then   engage to best suitor
        engaged(m)      true
        engaged(w)      true
        add (m,w) to list of engaged pairs
      else   w is already engaged
        if w prefers m to current proposer then
            w dumps m' and engages to m
          engaged(m')      false
          delete (m', w) from list of engaged pairs
          engaged(m)      true
          add (m, w) to list of engaged pairs
        else   w rejects m, so m remains unengaged
        endif
      endif
    endif
  endfor
```

Figure 2.1: The Gale Shapley Algorithm

2.4 Programming Tools

In order to develop CTMS, Microsoft Visual Basic.Net is selected as programming language and Microsoft Structured Query Language (SQL) Server 2000 for database programming.

2.4.1 Microsoft Visual Basic.Net

Visual Basic.Net was chosen for system interface programming. This is because Visual Basic.Net has a lot of advantages in term of object-oriented specification. Nearly all world-class software, from the leading Web Browsers to mission-critical corporate applications, is built using the Microsoft Visual Basic.Net development system. Visual Basic.Net is among the most productive tool for the highest-performance development for Windows.

2.4.2 Microsoft SQL Server 2000

Microsoft SQL Server 2000 was chosen as the database for this system. Microsoft SQL Server 2000 is the complete database and analysis that offers for rapidly delivering the next generation of scalable e-commerce, line-of-business and data warehousing solutions. Below are among the benefits of SQL Server 2000:

a) Fully windows application-Enabled – It can make query, analyze and manipulate data over the windows.
b) Highly Scalable and Reliable – It can grow without limits with enhanced scalability and reliability features. Partition database workload to achieve scale-out of applications. Take full advantage of Symmetric Multiprocessing (SMP) hardware.
c) Fastest Time-to-Market –Reduce development time with the integrated T-SQL debugger, and develop functions that can be reused in different applications. SQL Server 2000 provides the fastest route to windows application development.

CHAPTER 3

METHODOLOGY

3.1 System Overview

This project development is implemented based on a methodology and standard modelling technique in order to enable the communication becomes easier to understand, the system can be controlled and maintenance work can be done easier.

For that, this project is carrying out by using the Software Development Life Cycle (SDLC) method. It provides a consistent framework of tasks and deliverables needed to develop systems. The SDLC methodology is chosen as it includes only those activities that appropriate for this project. This is the most popular development model in the contemporary IT industry and most of the software products or systems have been developed successfully by using this model.

The SDLC is based upon two principles: dividing projects into phases, and using written documentation. While the exact phases in the cycle vary from one author or organization to the next, they generally follow along these lines, as shown in Figure 3.1: identify the system requirement, project initiation and planning, analysis, design, development implementation and testing. Each phase within the overall cycle may be made up of several steps.

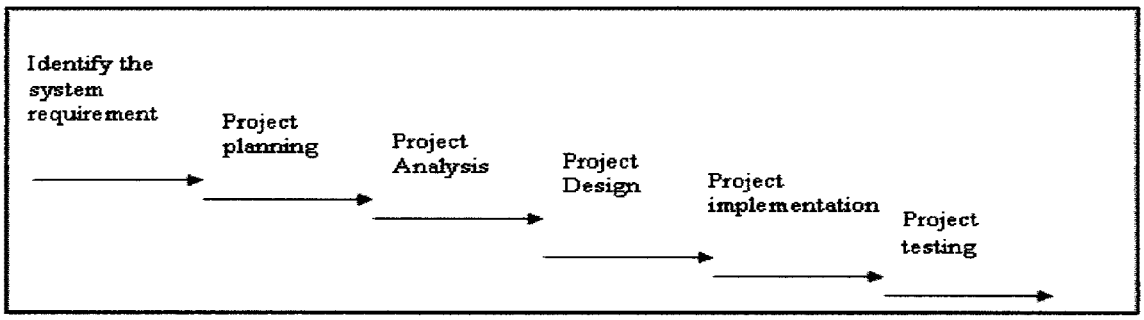


Figure 3.1: SDLC model

3.2 Identify the system requirements

The first phase of the development in which all the data and information required is identified, collected and analyzed. The purpose of this phase is to determine the needs and define the problems that need to solve. Thus, during the process, the possible information's that could undertake include:

a) Identifying the system of play and rules that used in chess tournaments in KUKTEM chess club.
b) Identifying the reason system of play used in chess tournament.
c) Identifying the limitation of each system of play.
d) Consideration on what type of system architecture will be used and what type of pairing algorithm applied in all chess tournament system

3.3 Project Initiation and Planning

In this phase, a high level view of the intended projects need to be establish and the goals is determined. It is a critical activity in the life cycle of project in which at this point, projects would be accepted for development, rejected infeasible, or redirected. The system initiation and planning process are explained in below section.

3.3.1 Initiation

a) This module is to do the entire pairing task for every round in chess tournament as well as allocate colours for players.
b) 50 students only are allowed to register as player to prevent the program from responding slow.

3.3.2 Planning

a) The Gantt chart (appendix A) shows the duration of each activity that being done during the project. The planning stage starts from propose the project title and defining the scope. Then, continue with the out come of the literature review and methodology in which is analysis process is the major activity.
b) Each module's task divided, system and module architecture diagram was designed to provide clearer view of the overall system.

3.4 Analysis

During analysis phase, the overall system and module architecture is studied and alternative replacement systems are proposed. There are three main activities in analysis phase: requirement determination, requirement structuring and alternative generation and selection process. Requirement determination is the only use in the development of this system.

Requirement determination is process of finding resources either primary or secondary resources. All data that required is collected from member in KUKTEM's chess club about the rules and limitation in chess tournament. Besides, pairing algorithm

and colour allocation method also collected from the books, internet and thesis documentation is analyzed to meet the requirement of the system.

3.5 Design

The physical characteristics of the system are designed during this phase. All interfaces are established and their function was defined. A design strategy is a combination of system features, hardware and acquisition method that characterize the nature of the system and how it will developed. Basically, the module design can be divided in to three which are:

a) Tournament Player Profile Management
b) Pairing Task for the entire tournament
c) Colour Allocation for every board in each round.

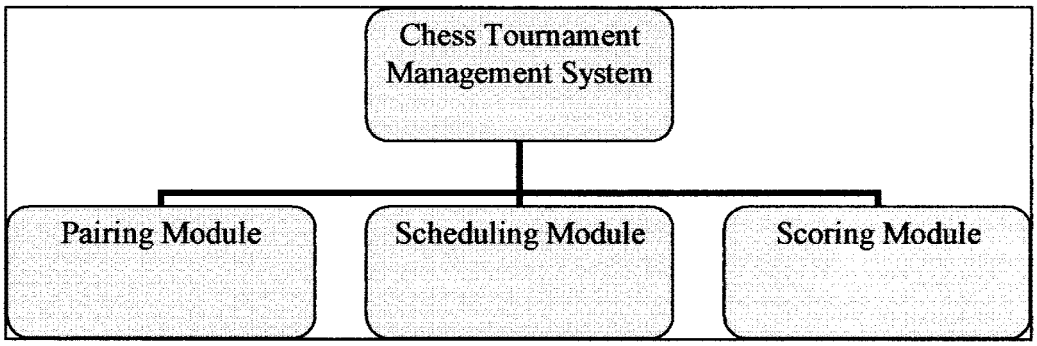


Figure 3.2: System Architecture Diagram

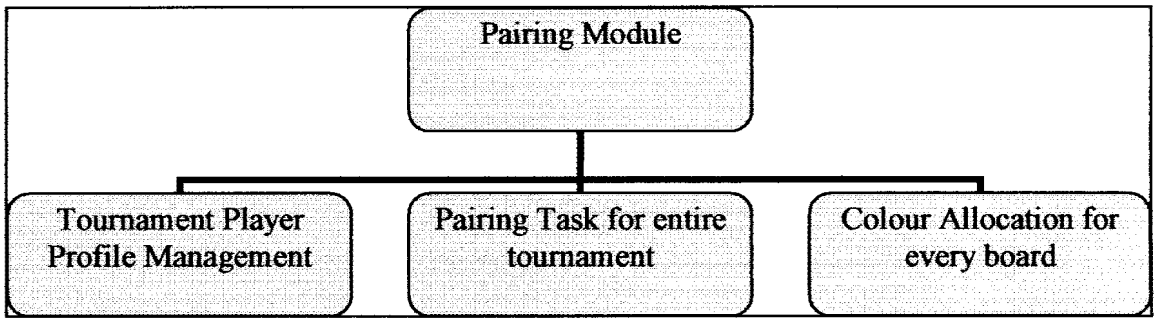


Figure 3.3: Module Architecture Diagram

3.5.1 Tournament Player Profile Management

This is a module that allows the system to do the registration part for all students that interested to join a tournament. Before they join a tournament, they first need to register as club member and tournament organizer will then key in their name and identification number. After that, organizer will drag their name to player's name if anyone of them wants to join the tournament. All players' detail will be set into database. Once they join the tournament, they will automatically get a club rating. The club rating represent the ranking of the player. If they continuously win in match, their rating will keep increased.

3.5.2 Pairing Task for entire tournament

Generally pairing task can divided into two because this system are using two type system of play which are Swiss System pairing and Knock Out Pairing.

3.5.2.1 Swiss System Pairing Task

Swiss System Pairing is a special pairing system suitable for chess tournaments with a large number of participants without eliminating any players in the process. Normally, as 1 player continues to win games, he or she will face progressively stronger opposition, leaving those not so successful to play against each other. Toward the end of the event, player will probably find himself matched against players around your own level.

Based on figure 3.4, for the first round, the players are paired either randomly or club rating. For randomly pairing all player will paired based on randomize list. For pairing based on club rating, the top player in the upper half of the field is then paired against the top player in the lower half of the field, and so on. The top-ranked player get white colour in the first round, and then colours alternate down the halves.

In the second round, the organizer uses the same principles to pair each of the three score groups (those who won and those who lost). These pairing procedures will continue through the rest of the tournament. Player with highest score gets the champion.

Proposed Swiss System Algorithm

1. Get total round and System of play in tournament
2. Sort all player based on club rating.
3. If no club rating was available, sort all players randomly.
4. Get new list of player.
5. Auto assigns number for all players.
6. Pair all player based numbers.

White		Black
Player Number 1		Player Number 2
Player Number 3		Player Number 4
Player Number 5	Vs	Player Number 6
Player Number 7		Player Number 8
Player Number 9		Player Number 10

7. Get result and get new player list.
8. Sort player again based marks and solkoff.
9. Assign number for all player based on marks and solkoff.
10. The round continues until end of round.

Figure 3.4: Proposed Swiss System Algorithm

3.5.2.2 Knock Out Pairing Task

In Knock out Tournaments, all players will be paired based club rating. Based on figure 3.5, all players in tournament will be sorted based on club rating. All players will be divided into two groups. The top player in the upper half of the field is then paired against the top player in the lower half of the field, and so on. Strong players will not meet each others in earlier round by using this method. The top-ranked player get white colour in the first round, and then colours alternate down the halves.

Player who win in the match will qualified to next round and paired with next player who win his match. This flow will be continued until the winner born.

Proposed Knock out Algorithm

1. Get total round and System of play in tournament
2. Sort all player based on club rating.
3. If no club rating was available, sort all players randomly.
4. Get new list of player.
5. Divide the list into group 1 and group 2
6. Auto assigns number for all players.
7. Pair all player based numbers.

White		Black
Player Number 1	Vs	Player Number 5
Player Number 2		Player Number 6
Player Number 3		Player Number 7
Player Number 4		Player Number 8

8. Get result and get remain player list.
9. Divide all winners into group 1 and group 2
10. Pair all winners.
11. The round continues until end of round.

Figure 3.5: Proposed Knock out Algorithm

3.5.3 Colour Allocation for every board

Colour is a very important manner in all chess matches. Player who gain white colour will gain more advantage compare to his opponent. To provide a fair treatment to all players, white colour will be given alternately to player. In order to do that, a record of white colour gainer must be saved.

3.5.4 SQL Server 2000 Database

SQL server database is place where this system uses for storing and organizing data especially in chess pairing and colour allocation process. There are three tables that operate in this module. The contents and explanation of each table as well the relationship are show in following table.

Table 3.1: NewTourTable content

Field	Type	Key	Default	Extra
id	int (10)	PRIMARY	NULL	auto-increment
name	varchar (30)			used
system	int (10)			used
round	int (10)			used

Table 3.2: NewTourTable description

Field	Description
id	Tournament identification number. Used to identify tournament. Act as primary key in this table
name	Tournament's name.
system	System of play which are either Swiss System or Knock Out.
round	Number of round to be play in the tournament.

Table 3.3: tour_play_table content

Field	Type	Key	Default	Extra
player_id	varchar (30)	PRIMARY	NULL	used
player_name	varchar (30)			used
player_no	int (10)			used
player_res	int (10)			used
tour_id	int (10)	FOREIGN	NULL	used
pre_color	int(10)			used
player_rating	float(4)			used

Table 3.4: tour_play_table description

Field	Description
player_id	Player's identification number. To identify player. Act as primary key.
player_name	Player's name.
player_no	Player number. Reference number that allowed the pairing algorithm to do pairing task.
player_res	Player's previous result in last match.
tour_id	Tournament identification number. Used to identify tournament.
pre_color	Player's colour in last match.
player_rating	Player's rating.

Table 3.5: PairingTable content

Field	Type	Key	Default	Extra
pair_id	int (10)	PRIMARY	NULL	auto-increment
tour_id	int (10)	FOREIGN	NULL	used
round_no	int (10)		NULL	used
board_no	int (10)		NULL	used
white	varchar (30)			used
Black	varchar (30)			used
white_res	int(10)			used
black_res	int(10)			used
white_no	int(10)			used
black_no	int(10)			used
white_rating	float(4)			used
black_rating	float(4)			used

Table 3.6: PairingTable description

Field	Description
pair_id	Pair identification number. To identify pair. Act as primary.
tour_id	Tournament identification number. Used to identify tournament.
round_no	Round number.
board_no	Board number.
white	White side player's name.
black	Black side player's name.
white_res	White side player's previous result in last match.
black_res	Black side player's previous result in last match.
white_no	White side player's number. Reference number that allowed the pairing algorithm to do pairing.
black_no	Black side player's number. Reference number that allowed the pairing algorithm to do pairing.
white_rating	White side player's rating.
black_rating	Black side player's rating.

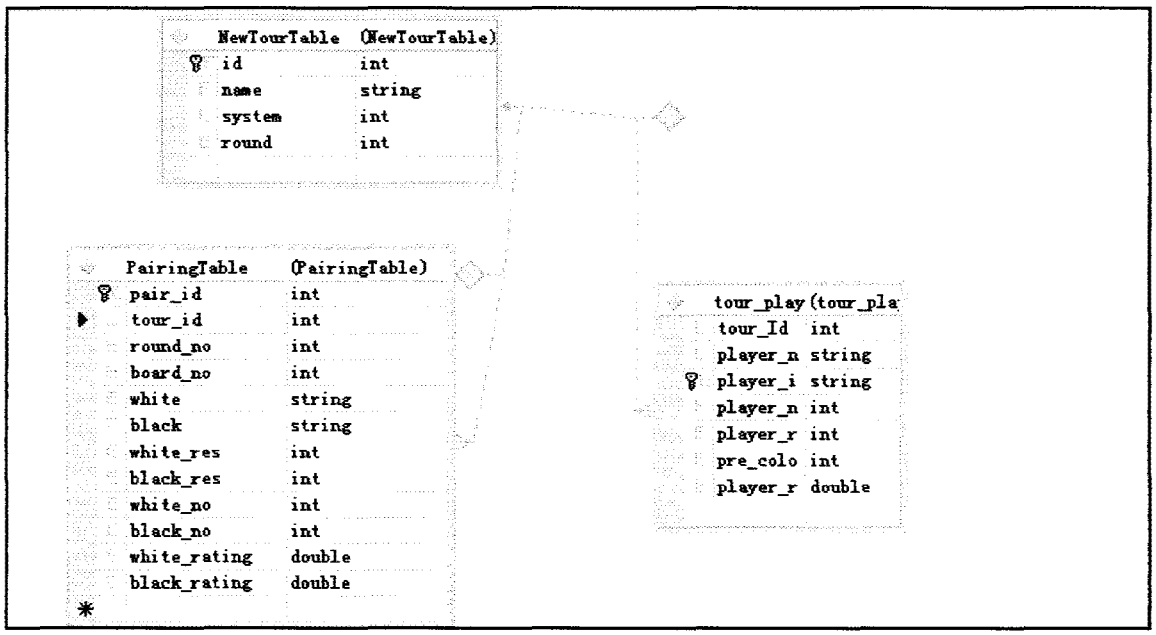


Figure 3.6: Relationship of Tables

3.6 Implementation

In implementation phase, the entire prototypes are developed and putting together to be tested. The physical specifications of the system are converted into reliable software. Finally, the application is tested to determine whether it is function properly or not.

3.7 Testing

The testing phase is a critical part in which the system that has been installed and configured is tested and examined to identify any error that occurred. There are many other aspects that need to be considered such as the response time, data limit and robustness during the testing process. Testing of system is done in step by step until the system can function properly. This phase starts with unit testing for both pairing and scoring modules. Both modules were tested step by step until each part of module function properly. System testing was implemented after both modules were combined. This type of testing is to ensure all parts in the system perform function perfectly even after both modules were combined.

CHAPTER 4

RESULT AND DISCUSSION

4.1 Testing Output

For testing output, features that will be test is the player profile management, pairing task and colour allocation.

4.1.1 Player Profile Management

a) Input: Figure 4.1 is the main interface of the chess pairing module. When <<New Tournament>>was clicked, New Tournament interface showed. Based on figure 4.2, a set of data was entered in New Tournament interface to create a new tournament. A message box was displayed as the data was entered showed a new tournament was created.

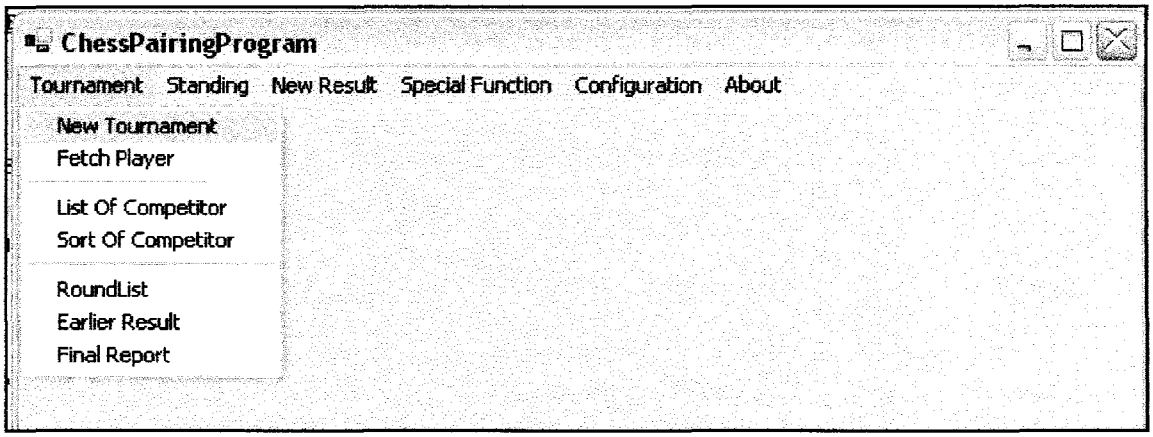


Figure 4.1: Main Interface

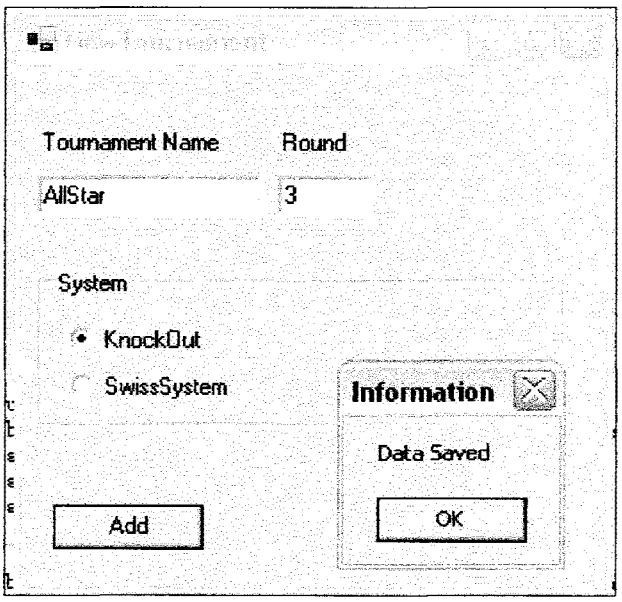


Figure 4.2: New Tournament Interface

Output: Based on figure 4.3, a new set of data was entered in NewTourTable

Data in Table 'NewTourTable' in 'CTMS' on '(local)'

id	name	system	round
26	god	1	3
27	AllStar	1	3
*			

Figure 4.3: Data in NewTourTable

b) Input: Based on figure 4.4, a new set of data was entered in Add Player Detail Interface to create a new player profile. A message box was showed to inform new player profile was saved.

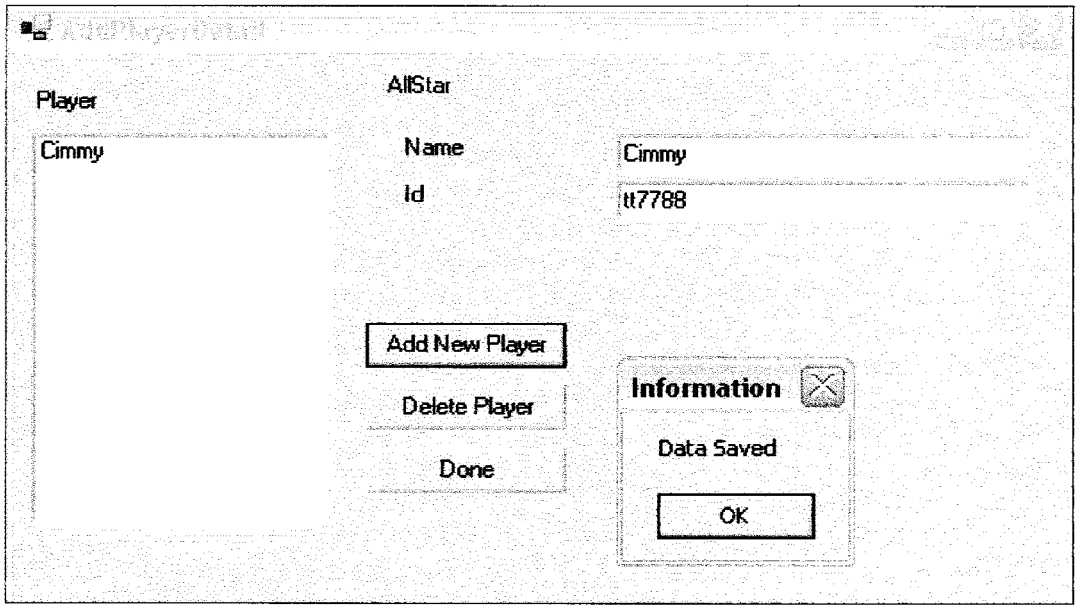


Figure 4.4: Add Player Detail Interface

Output: Based on figure 4.5, a new set of data was entered in tour_player_table by set the rating of player to 700 point as a starting rating.

	cheeyou	cb02077	26	700
	michael	cc1234	1	800
	tythfgh	fds	1	800
▶	[illegible]	[illegible]	[illegible]	[illegible]
*				

Figure 4.5: Data in Table tour_player_table

4.1.2 Pairing Task

a) Input: Based on figure 4.6, a list of player name was showed in Player Entry Interface. Five players was selected and saved by click on <<save>> button. A message box was showed to inform new player profile was saved. User will automatically linked to an interface called Generate Pairing Interface as figure 4.7.

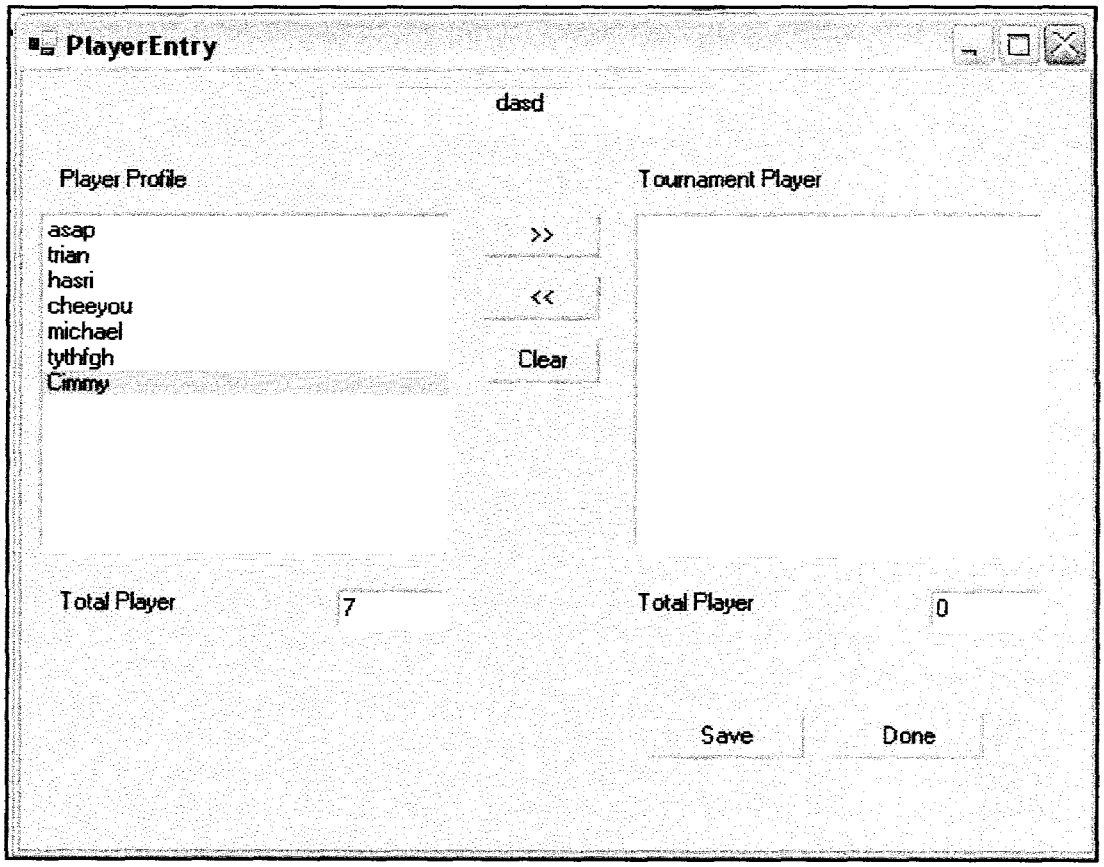


Figure 4.6: Player Entry Interface

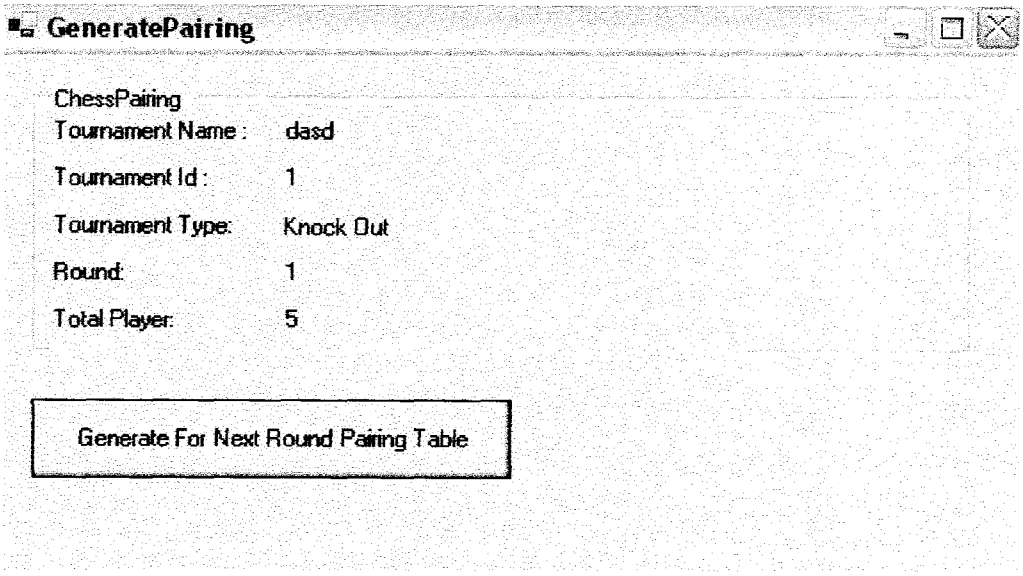


Figure 4.7: Generate Pairing Interface

Output: Based on figure 4.8, An Pairing table was generated with an auto allocation of board number and colour for each pair of player. Result need to be entered for next round pairing. The pairing task will be done for the rest of the round.

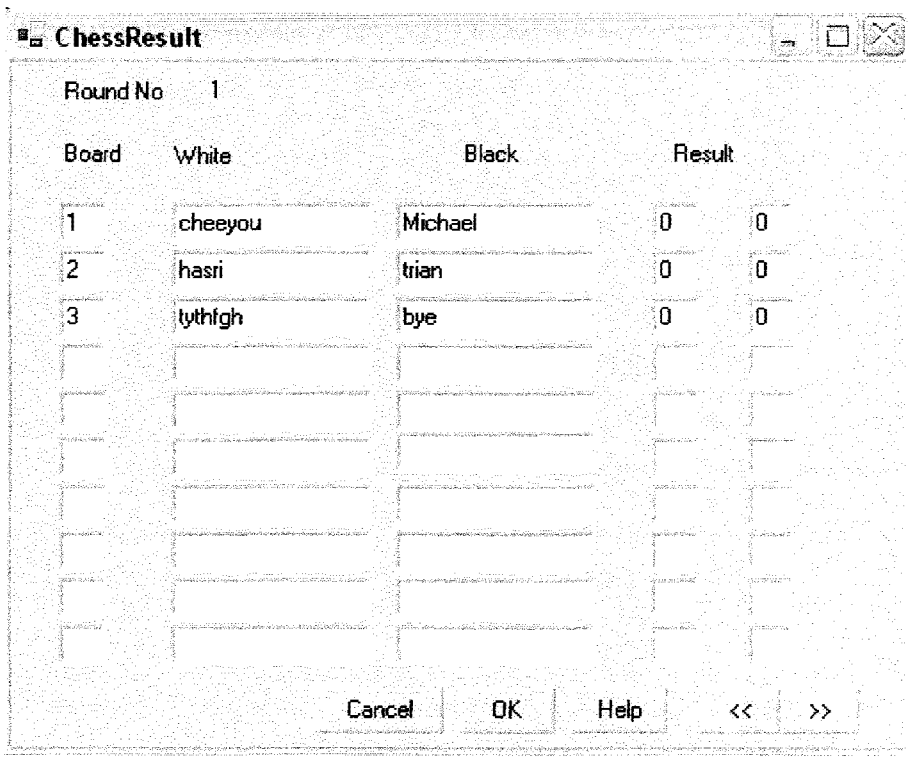


Figure 4.8: Chess Result Interface

4.2 Discussion

Based on output that generated through interface, this algorithm that based on Gale Shapley algorithm will first retrieve data from multiple table and do a complete sorting based on Solkoff point and rating. The entire sorting task was done by this algorithm, and all data in table will still remain the same. Then, this algorithm will check

for previous result and colour allocation before generate the pairing and colour allocation. Based on player entry interface, 5 players were entered into the system. All players got their own rating. These 5 players will gain an auto generate reference. They will divided into 2 group , player number 1 will pair with player number 3 , follow by player number 2 with number 4 , and the flow continue until player number 5 got no one to pair with. At this stage, this player got a bye. Based previous colour record, player will got an alternate colour in every match.

4.3 Advantages

This algorithm will sort all players based on Solkoff point and rating and generate paring based on system of play that used by that particular tournament. Players with high rating will not meet each other in earlier round. Besides that, colour allocation was given to players alternately to provide to fair treatment to all players since colour can be a big advantage for all player. There will be no complaint about unfair treatment in that particular tournament because all jobs were done automatically.

4.4 Constraint

The major constraint in this module is the problem of handling more than 50 players in a tournament. This module is able to perform pairing task for a tournament in less than 1 minute if less than 50 tournament's players was entered. Once amount of players more than 50, the overall process of pairing will get slower. The pairing task can even take time more than 3 minutes if players were more than 100.

4.5 Further Research

This pairing module uses two system of play that most commonly used in all chess tournament, which are Swiss System and Knock out Pairing. Further research should be done in applying more systems of play to this module. It is wise to provide more option for players so that this system can be used in larger tournament. A complete study in international chess tournament rules and limitations need to be done in order to provide more fair treatment to players. Besides that, further study need to done especially in handling more players and larger size of tournament. The whole pairing task should be done without affecting the system performance.

CHAPTER 5

CONCLUSION

Chess Tournament management is a very complicated task that need to be handled by tournament organizer. A lot of time needed to complete that entire pairing and colour allocation task in chess tournament; participant will need to wait for a long time until the overall pairing table published. Particularly the algorithm used to solve the pairing and colour allocation in this module was promising. Pairing table can be generated right after each chess board result was entered. It helps reduce time spend and mistakes that might occur in chess tournaments.

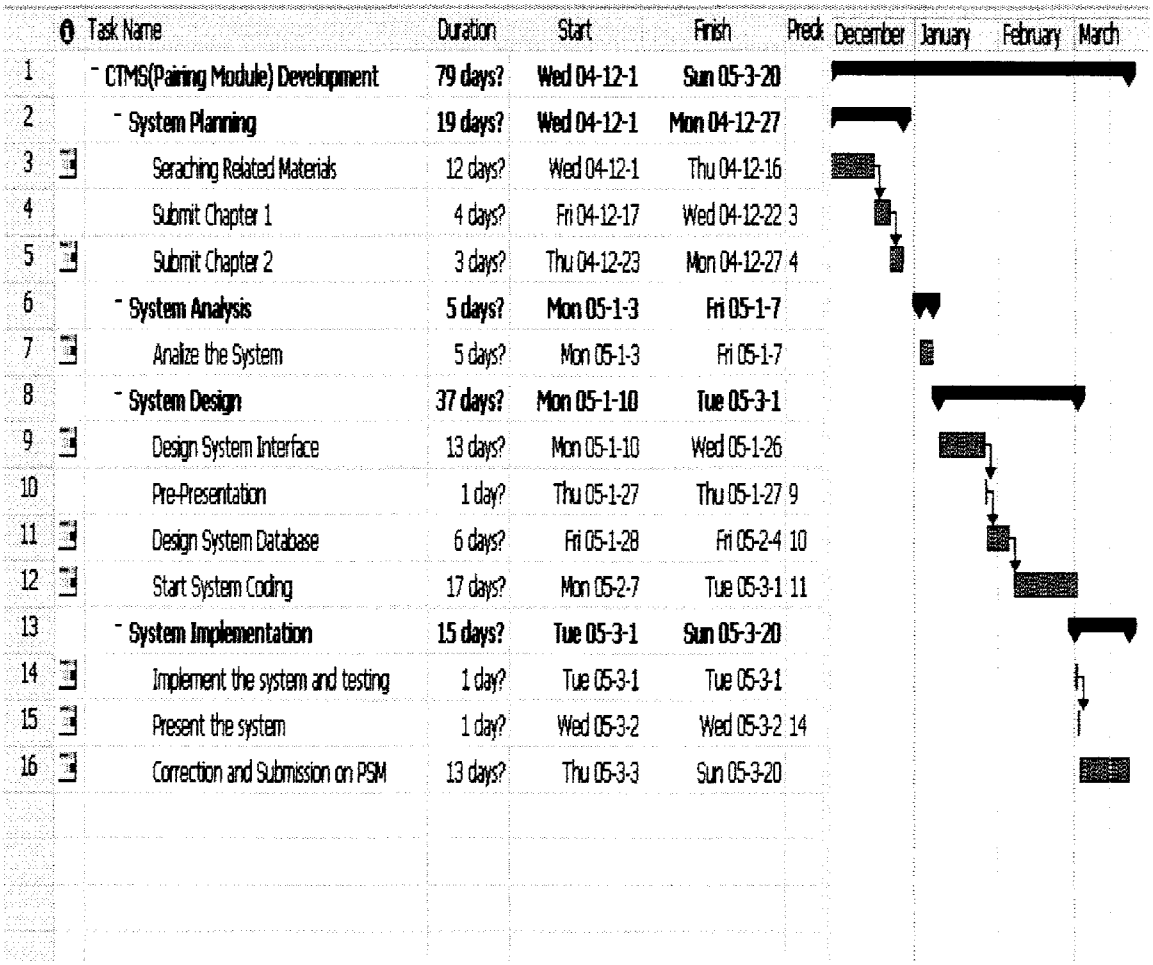
However, the question raised is how to enable the module to handle more than 50 players without making the whole pairing process heavy and slow. Further research should be done in applying more systems of play to this pairing module that used in international level chess tournament by studying in their rules and limitations.

REFERENCES

1. Gus_eld, D., *The structure of the stable roommate problem: efficient representation and enumeration of all stable matchings*, SIAM J. Comput. 174 (1988), 742-769
2. Gus_eld, D., Irving, R. W., *The Stable Marriage Problem. Structure and Algorithms*, The MIT Press, 1989.
3. Hooper, D., Whyld, K., The *Oxford Companion to Chess*, Oxford University Press, 1984.
4. Irving, R. W., An Efficient Algorithm for the "*Stable Roommates*" Problem, J. Algorithms 6 (1985), 577-595.
5. Olafsson, S., *Weighted Matchings in Chess Tournaments*, J. Opl. Res. Soc. 41, 1(1990), 17-24.
6. Krause, Ch., Protos, Version 6.ENG.-*A computer Program for the Swiss Pairing System*, 1994.

APPENDICES

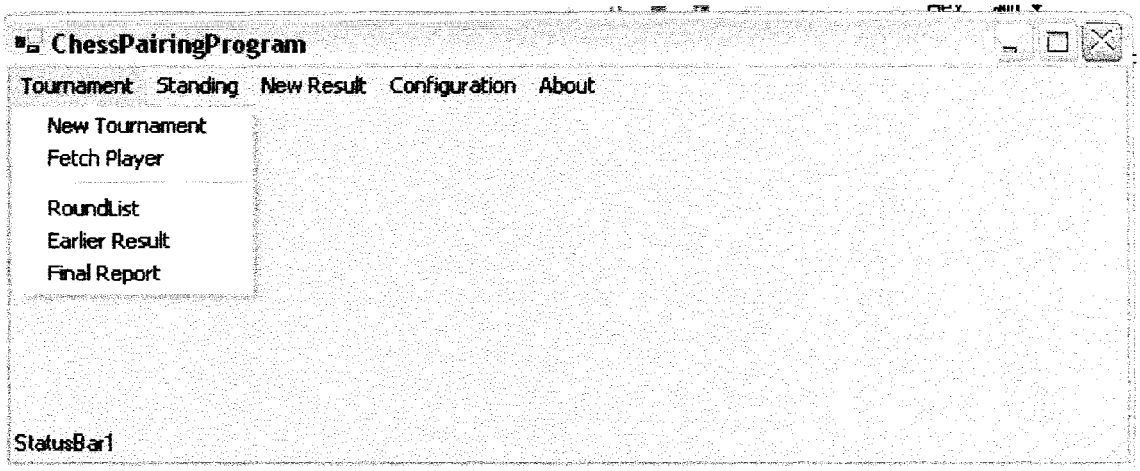
Appendix A: Gantt chart



	Task Name	Duration	Start	Finish	Predc
1	**CTMS(Pairing Module) Development**	**79 days?**	**Wed 04-12-1**	**Sun 05-3-20**	
2	**System Planning**	**19 days?**	**Wed 04-12-1**	**Mon 04-12-27**	
3	Seraching Related Materials	12 days?	Wed 04-12-1	Thu 04-12-16	
4	Submit Chapter 1	4 days?	Fri 04-12-17	Wed 04-12-22	3
5	Submit Chapter 2	3 days?	Thu 04-12-23	Mon 04-12-27	4
6	**System Analysis**	**5 days?**	**Mon 05-1-3**	**Fri 05-1-7**	
7	Analize the System	5 days?	Mon 05-1-3	Fri 05-1-7	
8	**System Design**	**37 days?**	**Mon 05-1-10**	**Tue 05-3-1**	
9	Design System Interface	13 days?	Mon 05-1-10	Wed 05-1-26	
10	Pre-Presentation	1 day?	Thu 05-1-27	Thu 05-1-27	9
11	Design System Database	6 days?	Fri 05-1-28	Fri 05-2-4	10
12	Start System Coding	17 days?	Mon 05-2-7	Tue 05-3-1	11
13	**System Implementation**	**15 days?**	**Tue 05-3-1**	**Sun 05-3-20**	
14	Implement the system and testing	1 day?	Tue 05-3-1	Tue 05-3-1	
15	Present the system	1 day?	Wed 05-3-2	Wed 05-3-2	14
16	Correction and Submission on PSM	13 days?	Thu 05-3-3	Sun 05-3-20	

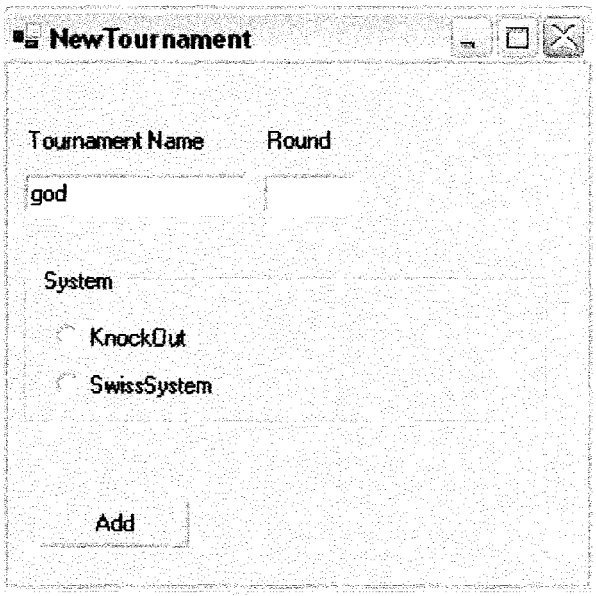
Appendix B: User Manual

Figure A: Main Interface



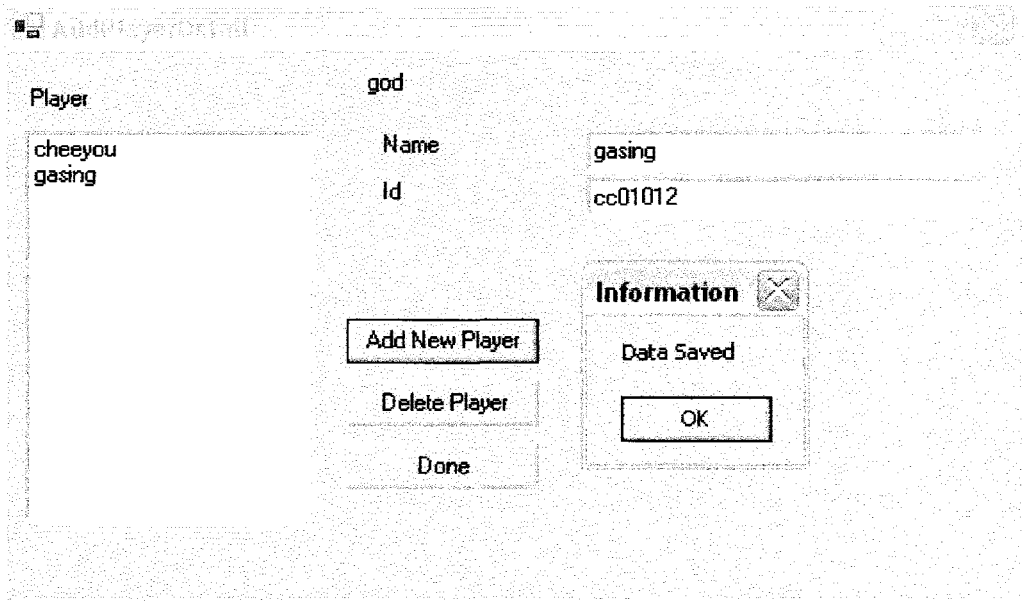
When we come to chess pairing program, the main interface as figure A will showed. The first thing user need to do is click on new tournament at menu to organize a new tournament.

Figure B: New tournament interface



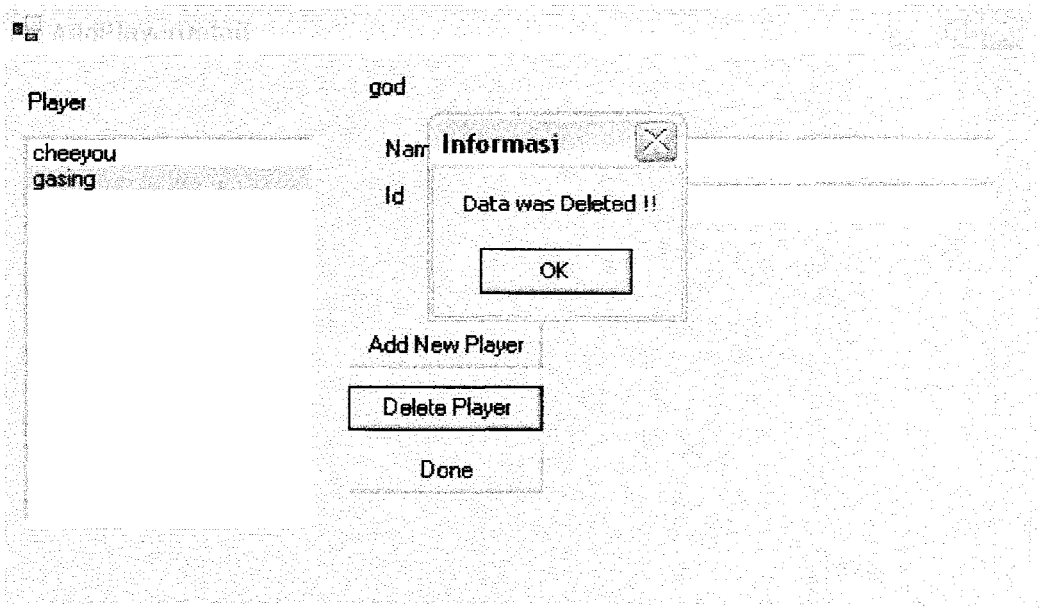
Next, new tournament interface B will be showed. User need to key in tournament detail including what tournament name, how many rounds to play and system of play.
Then click on add button.

Figure C: AddPlayerDetail interface



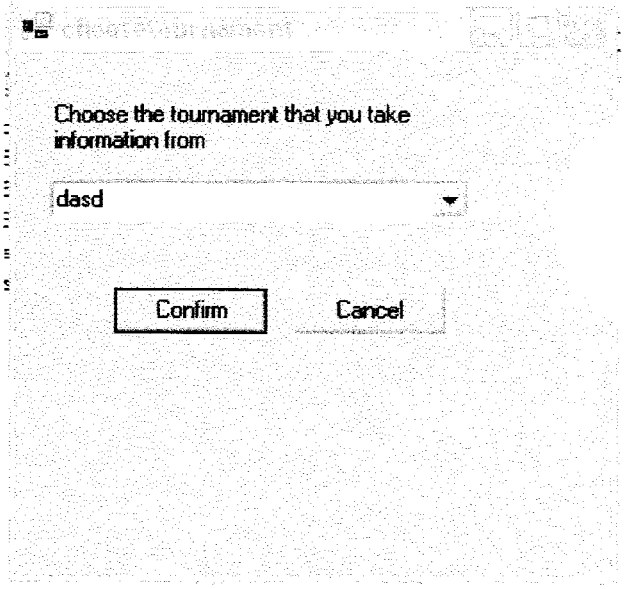
Then, add player detail interface as figure C will be appeared. In this interface , if user want add new chess club member, then they can click on <<Add new player >> button to add new player for chess club. Once the button was clicked, Messagebox "Data Saved " showed.

Figure D: Add Player Interface



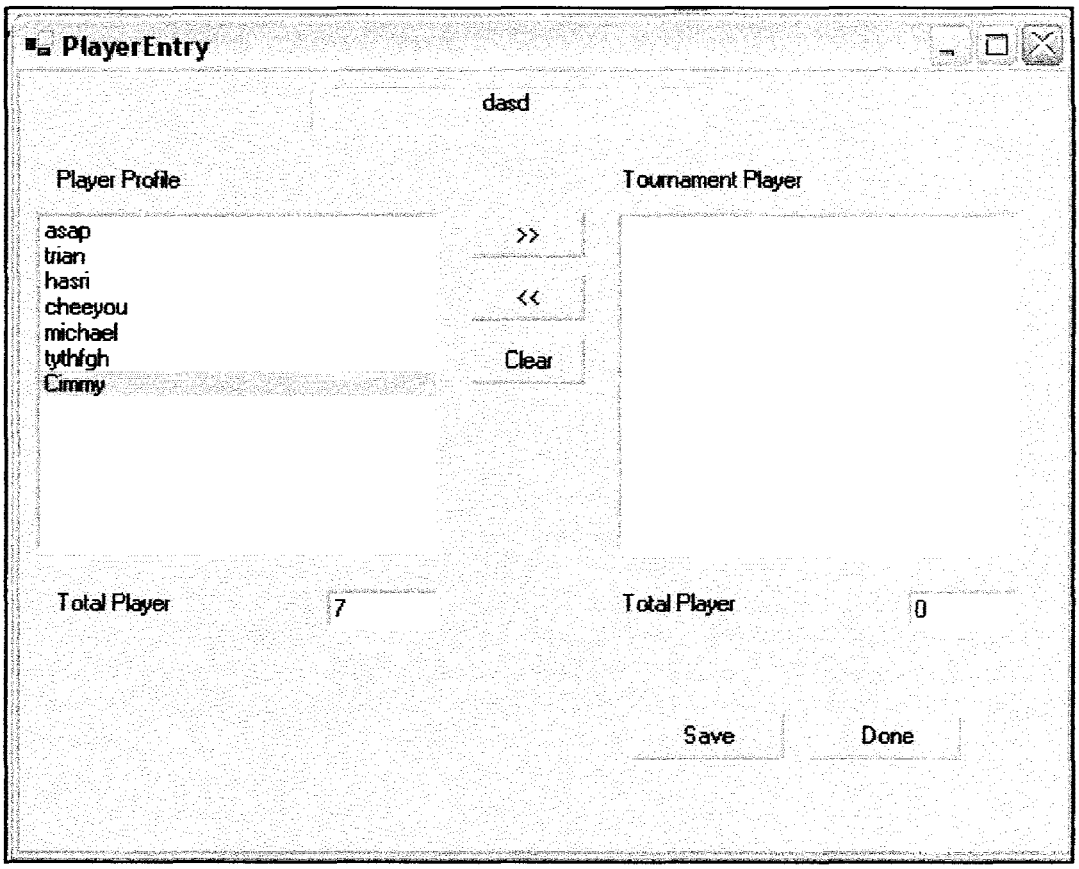
If user wants to delete the record they add previously, they can click on <<Delete Player>> to delete the name. Once the button was clicked, Message box "Data was deleted "showed.

Figure E: Choose tournament Interface



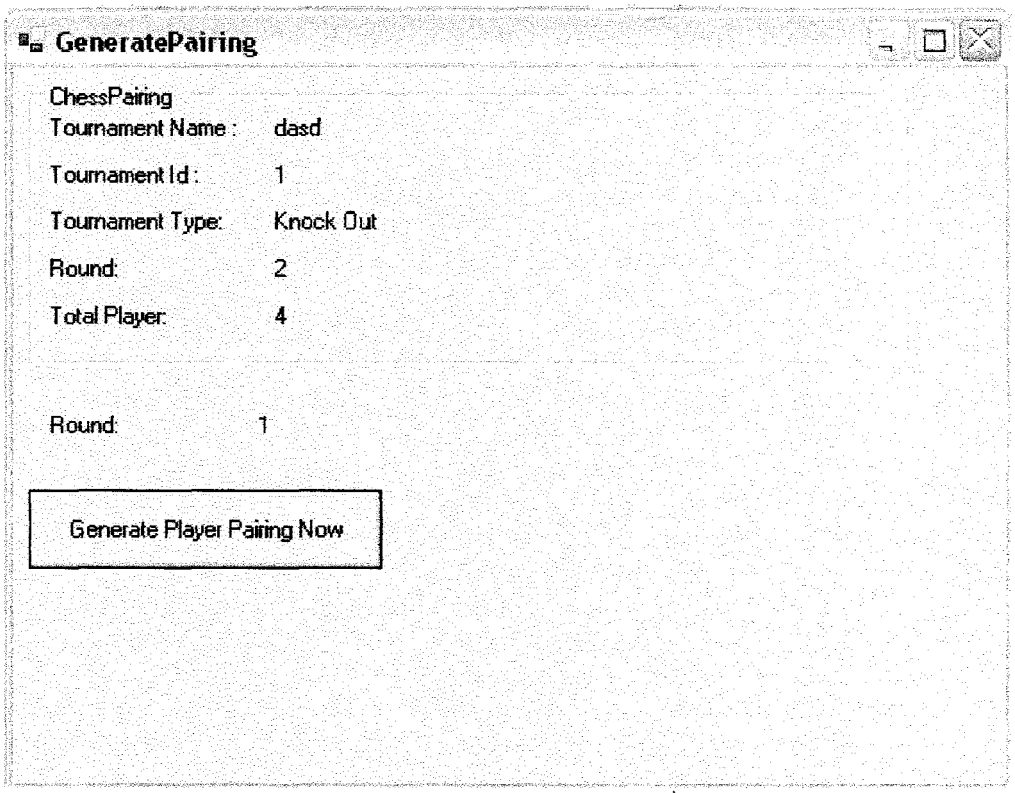
When user click on <<fetch player>> at main interface, choose tournament interface as figure will showed. Here user needs to select tournament name that retrieve from database, and click on << confirm>> button.

Figure F: Player Entry Interface



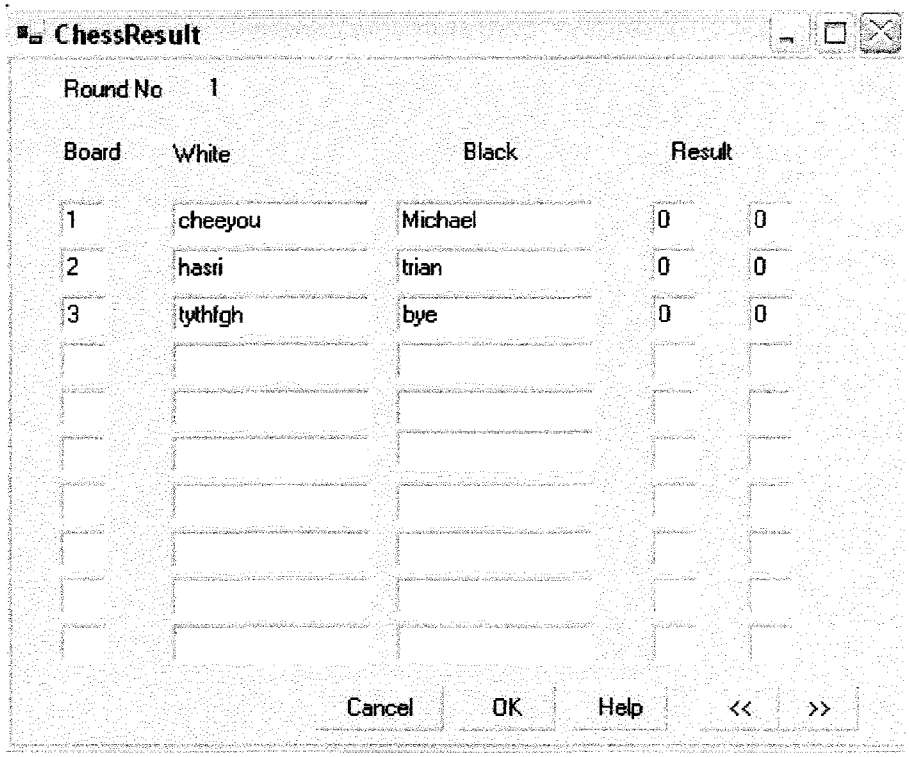
Once the user selects the tournament, user can now select which player he want to put in the tournament by highlight the name of player in player profile , then click on " >> "button to drag the player to tournament player. User can move it back by click on "<<" button. << Clear >> button is to clear the list in tournament player list.
After select players, user shall click <<save>> to confirm the selection.

Figure G: Generate pairing interface



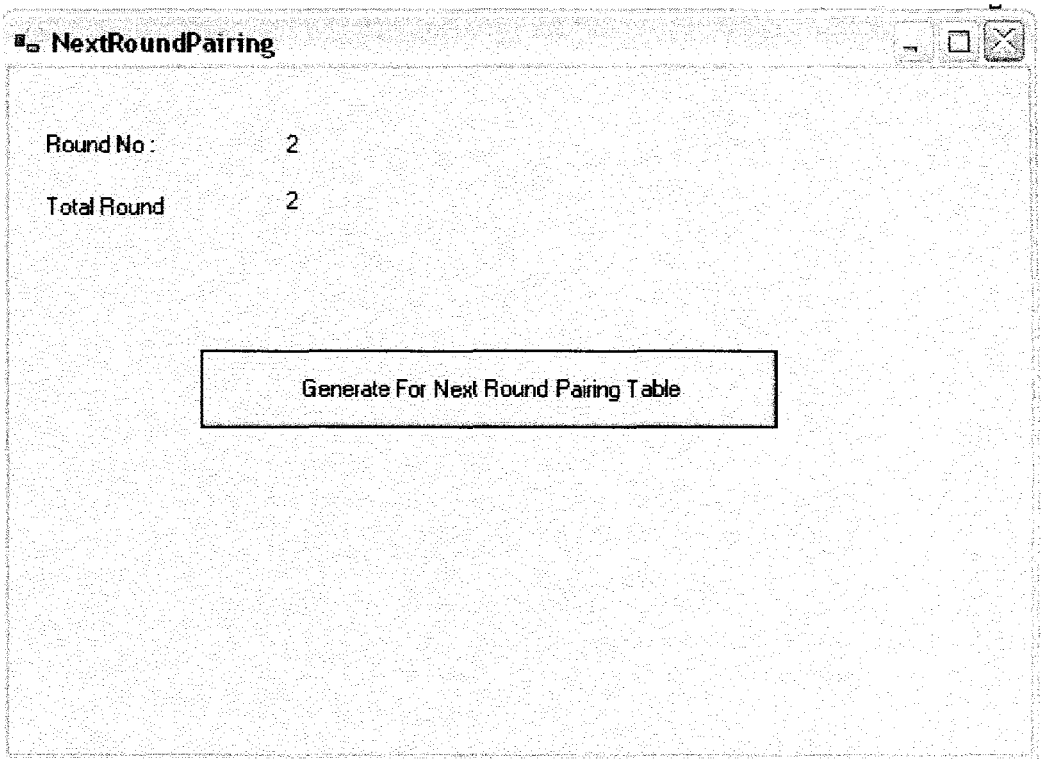
Once all tournament players were selected, to generate 1st round pairing, we shall go to generate pairing interface. Here, all detail of selected tournament will showed as figure G. User need to click on << Generate Player Pairing Now >> to generate the pairing for all players.

Figure H: Chess Result Interface



The next interface appeared will be as figure H which is the chess result interface. In this interface, all pair that generate before this will be showed. This interface allowed user to key in the result for each board.

Figure I: Next round Pairing interface



The last interface in this module will be as figure I which is an interface to do pairing for the rest of round in that tournament.