

Collecting statistical data in basketball

B. Markoski*, M. Ivković*, P. Vasiljević**, P. Pecev*** and Z. Milošević****

* Technical Faculty "Mihajlo Pupin", Zrenjanin, Serbia, markonins@yahoo.com, misa.ivkovic@gmail.com

** Faculty of Technical Sciences, Novi Sad, Serbia, petarv@uns.ac.rs

*** Dept. of Mathematics and Computers Sciences, Faculty of Sciences, Novi Sad, Serbia, predrag.pecev@gmail.com

**** Faculty of Sport and Physical Education, Novi Sad, Serbia, zorana@eunet.rs

Abstract - In last few decades sport has become more than just a game. Today it's a big business, with lots of money invested in it. Getting right information as quickly as possible makes a difference between winning or losing. Statistics in sport is most valuable to coaches, but also the whole population of sport lovers. Using modern technology and mass media like TV and Internet, people are able to follow their teams and players. This paper describes application used to collect statistical data during basketball game in Firebird relational database, programmed in C++ programming language. Presented solution is, with small changes, applicable to other sports.

I. INTRODUCTION

At the beginning of a season, coaches are mostly interested in using different statistical reports for analyzing and evaluation of individual players. Once they have insight into advantages and flaws of their players, their interest is moving towards the team as a whole. They want to know how good a team is. The team statistics therefore becomes most important. After all, a basketball is still a team sport. And finally, having in mind that different statistic reports may be used also for analyzing opponent's play, during a season coaches' interest is moved towards their opponents. More often than not, well analyzed opponents' play means a difference between winning and losing [1].

Coaches are not the only ones to use statistics. On the contrary, whole population of sport fans and audience is able, using mass production of technology and media as TV and Internet, to follow efficiency of teams and individual athletes. To many people, as reporters and commentators, statistic is an important tool in doing their job, and for some, as sport managers, it is a vital part of their profession. After all, during last decades sport became more than a game: it is a large business, with considerable amounts of money invested [2].

II. ANALYSIS AND DESCRIPTION OF SYSTEM

A. Basic Information About Basketball

Basketball is a team sport. Actors of a basketball game are the players from a two opposing teams, their team officials with coaches as well as officials (commissioner, referees, table officials, statisticians). Every team may have no more than 12 players per game, and of those 5 are actively engaged in every moment.

Regular basketball game is divided into 4 quarters of 10 minutes each, and if a result is draw, after regular time additional time of 5 minutes is played, as many times as necessary to decide a winner of a game.

There is no limitations regarding number of substitutions of players during a game, but there is limitation regarding personal fouls. If a player gets fifth personal foul he must leave the court and may not play any more.

A basket scored gets 2 or 3 points, depending on a distance from which the ball has been thrown. A line discerning two cases is drawn on the court [3].

B. Standard Statistic Elements in Basketball

Basketball is highly dynamic, and during a game a number of events may be seen on court. Since the aim of this paper is consideration of collecting and processing standard statistic elements in basketball, it is necessary to define those elements. In this purpose a statistic manual of FIBA (International Basketball Federation) was used [4]. It lists following elements of game: successful free throw, unsuccessful free throw, successful 2-point throw, unsuccessful 2-point throw, successful 3-point throw, unsuccessful 3-point throw, defensive rebound, offensive rebound, assist, steal, turnover, foul, technical foul, block, play time.

By collecting data regarding those elements, it is possible to generate most of the standard statistic reports. But, in order to have complete statistics with additional statistic reports, it is necessary to add certain elements [5].

Possibilities for adding to certain elements of game are:

- For throws, successful or not, for 2 or 3 points, it is possible to add information from which position on the court a throw was made.
- For turnover it is possible to add information regarding a way that ball was lost (poor dribbling, poor pass...).
- For fouls, it is possible to note on which player the foul was made.
- For blocks, as well as for fouls, it is possible to note on which player the block was made.

C. Problem of Tracking Time

Tracking time is a problem that requires special consideration. Is there a reason to keep track on time, and how precise it must be? The fact is that basketball periods (quarters and additional time) are precisely defined but it is not the reason to keep track of time, since beginning and end of the period may be defined as separate actions and to be noted as such.

The main reason for including time is a need to obtain information how much time certain player spent on court. Without this information, efficiency of a player may not be measured by other statistic element alone. For instance, by comparing number of points for two players we may conclude that the one with 20 points scored is better player (or at least that he played better at the game observed) than the one who scored 10 points. But, if we have additional information that first player played 40 minutes and the second one only 5, conclusion is completely different [6]. There are other reasons to introduce keeping time, for instance need to note in which parts of a period certain actions occurred, or possibility to separately analyze only certain periods of game, etc [7].

Regarding precision, our opinion is that it is quite sufficient to express time in minutes. For this purpose another action must be added: user may enter scoreboard time after every minute of the game. In this way, there is possibility to correct time intervals. In case when more precise information on time is needed, it is possible to re-define such action to shorter intervals (for instance 15 seconds).

D. Structure of Actions

In order to collect statistic data, coding must be done, i.e. every statistic element must be given unique numerical code. In other words, user collects statistic data by evidencing relevant events, entering following data for each: which player was involved in particular event, which action it is (a code of statistic element for this event and, depending on element evidenced, additional data - throw position, type of turnover etc.). Such set of data entered for observed event is called action. Therefore an appearance of an action may be specified.

Action comprises of 4 elements:

A0 – team mark (1 - host, 2 - guest)

A1 – number of player's shirt

A2 – code of event (statistic element)

A3 – code for addition to event

E. Choosing Periphery Unit for Data Collection

There are two standard periphery units that enable user to input data: keyboard and mouse. Both have strong and weak sides. Strong sides of one are the weak sides of other, and vice versa.

Keyboard input demands user to know event codes, while for mouse input he may not know codes: it is enough to click on button with event name. But, if user knows event codes, and especially if he has routine using numerical pad on keyboard, this way of data input is

considerably quicker than using mouse since there is no time lost on precise positioning the cursor. Another advantage in this method is that user may pay more attention to events on court instead constantly watching monitor while using mouse for input.

Since number of possible events and their codes is not too large, authors decided to realize data input using keyboard. It is also possible, if the need arise, to expand system for mouse input too. In such case, a user interface for mouse input is available, while functionality simply amounts to calling realized functions for keyboard input while clicking individual keys.

F. System Description

Aim of this system is partly to generate reports after game is finished. This may not be directly after the game is over, but also in a consequent period. Therefore, one of functions necessary for the system is a permanent storage of information. This is accomplished by storing information in a database. After every input by user, action is saved into a database. In this way a potential problem of power shortage during a game is solved or any other problem of a similar nature.

By analyzing other aims of this system and having in mind all that is presented in this section, we will give a short review of functions that a system should have:

- Collecting statistic data by keyboard input of actions by user. Input must be quick and precise.
- In order to enable input of actions, there must be possibility to create in advance a new game, including definition of basic data about it.
- Possibility to open existing game from a database, with potential proceeding of new actions input or changing basic data regarding the game.
- In all these cases, possibility to generate (show and print) dynamic statistical reports. Word "dynamic" means possibility to generate report at every moment during a game (both during and after a game).
- Possibility to update data about all actors in a game from database. Updating means possibility to add new actors, change data about existing actors and deleting actors.
- User friendly system with textual and graphic forms of help.
- System stability and precision in processing and presenting data is understood, and it is also desirable that a system is easy to use.

III. APPLICATION

A. Conceptual Data Model

In analyzing the real system, following object classes were found: game, league, club, coach, player, position, commissioner, referee, roster and action.

Object classes from a real system were copied one-on-one into objects of information system. Every object class is represented as a table in data model, and properties of a table represent characteristics of corresponding object class in a real system [8].

Conceptual data model was realized using CASE tool PowerDesigner as shown in Fig. 1.

B. Use Cases

Use cases [9] for the application may be divided into two basic subsets: 1. Updating of data in database, and 2. Keeping statistics of a game. Updating of data in database encompasses cases of updating leagues, clubs, players, coaches, commissioners, referees and statisticians. Keeping statistics of a game encompass cases of creating new game and opening existing one, and those comprise of collecting data and generating a report.

The diagram of use cases is shown in Fig. 2.

C. Database Implementation

Using PowerDesigner tool, a translation was done from conceptual data model into a physical data model compatible to chosen system for database management (Firebird).

On the basis of physical data model, PowerDesigner generates file "crebas.sql" consisting of SQL scripts for creating complete database, with all tables and their interconnections.

Using Firebird console, which together with Firebird server is a part of Firebird package, as well as using SQL scripts generated by PowerDesigner, a physical database have been designed [10].

Communication between application and database is done using Firebird server. Therefore, Firebird server is a necessary link between application and database, and it must be installed and running while system is in use.

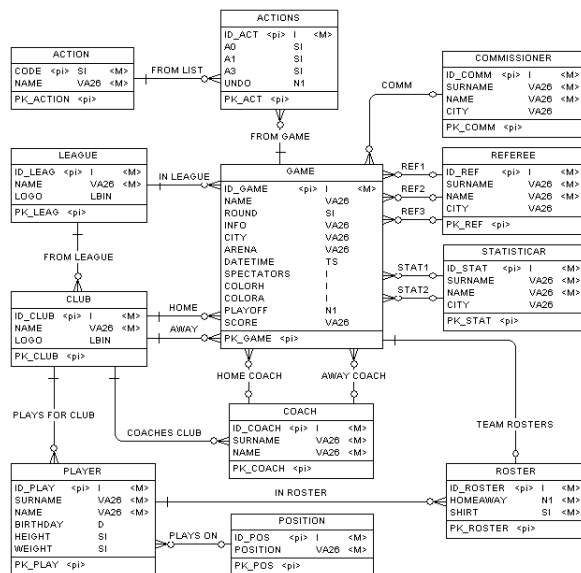


Figure 1. Conceptual data model

D. Functionality and Graphic Interface

In this section of paper functionality and graphic interface of application are shown on example of keeping statistics at a basketball game.

Before the game, statistician collects basic data regarding the game, as well as team rosters. After obtaining all necessary information he starts the application. When application is started, a basic window with a menu is shown.

Option "Game" has items for starting functions: to create a new game, open existing game, and leaving application. Items in option "Data" start functions of updating certain data within database.

Item "Settings" shows a form to update different options within application. Items of "Help" option show user help, as well as basic information about the application.

After application has been started, statistician may, if a need arises, correct settings of the application. In this way it is possible to set different parameters of report, environment, report graphic and security.

Statistician, using an application part for updating database, may check data from it and change or add new data into database (new league, clubs, players, coaches, commissioners, referees or statisticians).

After checking and updating database, statistician is ready to create a new game. After starting this function, a form for input game data is being shown.

Statistician confirms accuracy of input data by pressing "Confirm" button, and after that a game window (game form) is being shown. This window is, in a way, the main frame of the whole application since statistician spends most of the game working within it, entering actions and generating reports. Game window is shown in Fig. 3.

Under "Game" option there are items for changing game data, defining roster and lineups, as well as for exit

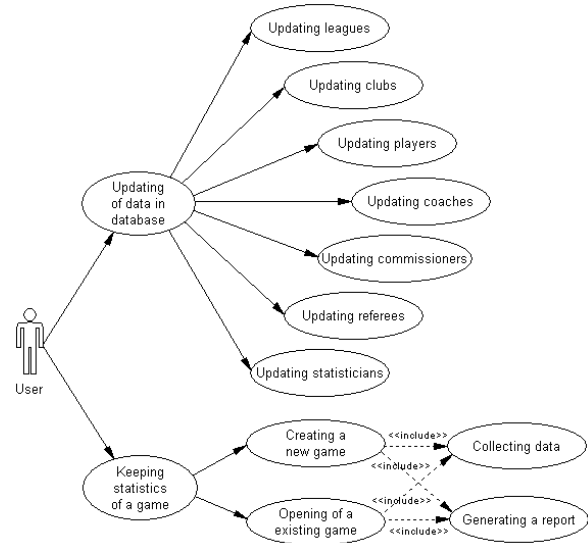


Figure 2. Diagram of use cases

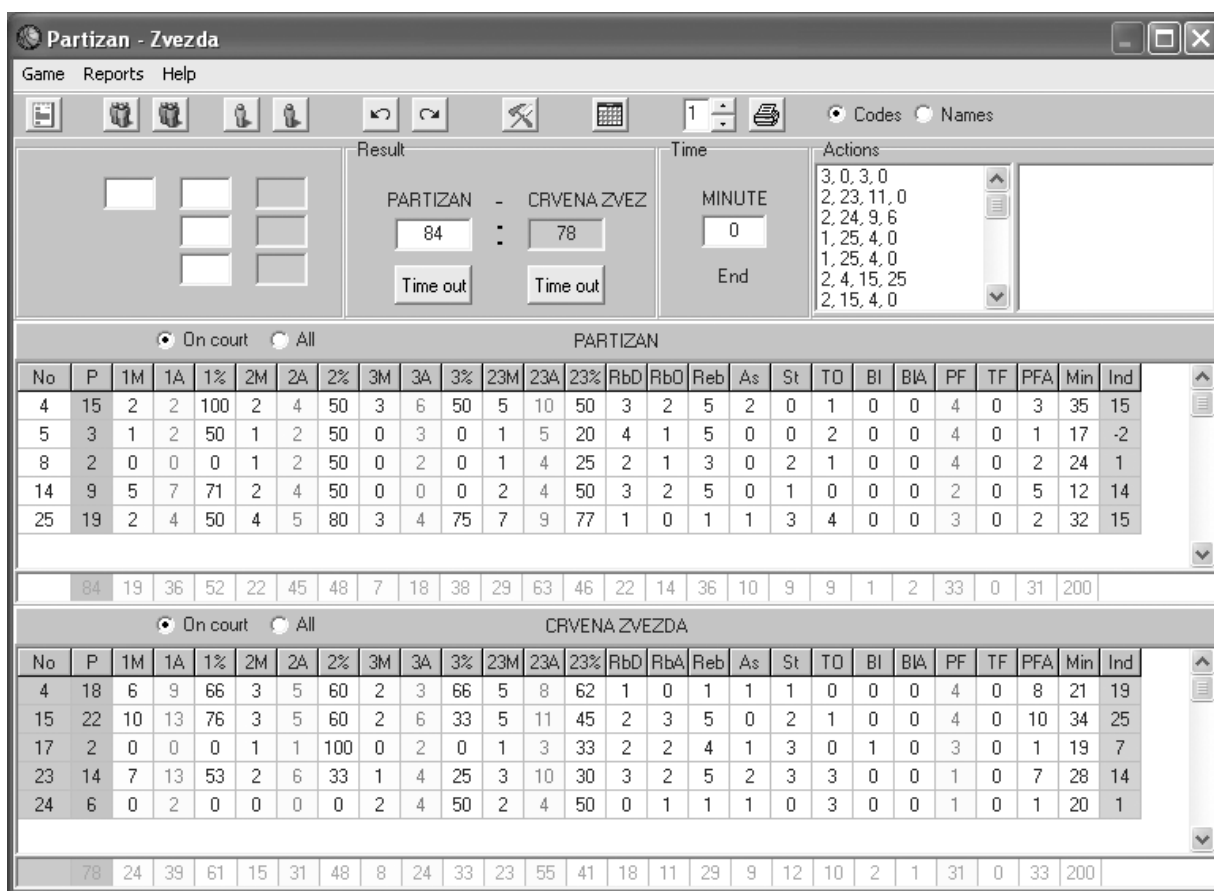


Figure 3. Game window

from current game. Items for "Reports" option start functions for generating (showing and printing) corresponding reports. Items from "Help" option show help options, quick review of actions and positions on the court, as well as possibility to assign certain colors to teams in order to facilitate management within a game window.

After creating a new game and entering all relevant information, statistician begins to define roster of both teams, entering shirt numbers for all players on a list, and defines team coaches.

Immediately before starting the game, statistician is informed about lineups for both teams and defines those in application.

From this moment on, statistician will follow events on court and enter appropriate actions in appropriate fields. All actions entered by statistician are shown in a list of input actions. If statistician makes a mistake when entering action, there is possibility to correct it. List of input action is an Undo list at the same time. By clicking the Undo button, last action entered will be deleted from the list and removed to Redo list. By clicking the Redo button, last action from Redo list is removed to list of input actions. Deleting is also possible by double-clicking the action and confirming delete.

Statistician has at his disposal a quick help in form of review of possible actions and court positions.

Most of realized statistic reports is possible to print in any moment during a game. Nevertheless, printing reports is usually done in following way. Before the game, after entering data about the game and the team rosters, report "Game announcement" is printed and distributed to coaches, media and managers. At the end of every quarter or additional time, "Basic report" is being printed for coaches, and in half-time this report is distributed to media and managers, even to players if required. If a coach during a game demands any report it will be printed for him. After the end of a game, a report is distributed to all actors, while coaches get all possible reports from a game. In this order a form for printing all reports is realized, so the statistician would not have to open one report at a time in order to print.

IV. CONCLUSION

After testing application implemented on a number of basketball games, conclusions are as follows:

- Action input using keyboard is fast and precise.
- Generating statistic reports is practically instant, so coaches may obtain timely information at every moment during a game. Data in reports are correct.
- Application is stable, and due to realized functionality of backup database, it is resistant to

possible unexpected situations during a game, as power blackouts etc.

- Application is very simple to use. With short training and trial work at only a few games, users are fully prepared to use application and to work on their own.

Implemented application fully realized aims set at the onset, as well as functionality by specifications in system description.

Implemented application is conceived to be open for further improvements and expansions.

Some of improvements possible are already mentioned in a paper, regarding widening set of statistic elements, corrections of time intervals, realization of interface for mouse data input and generating new statistic reports. Additional improvements could be found in an error-avoiding part in case of wrong data input by user. System offers action delete, and further improvements would be possible regarding changing already entered actions, inserting new ones between already existing ones and changing sequence of entered actions.

System expansion could be about collecting statistic data from several games and generating a general report for all player, clubs, and leagues. It is also possible to send joint statistic data to internet after the end of a game, so

they are available to more people. Availability may be obtained by connecting to graphic data-showing systems in TV broadcast stations.

Finally, it must be noted that a whole system, with certain changes and adjustments, may be applied to other team sports.

REFERENCES

- [1] S. Trninić, "Analiza i učenje košarkaške igre", 1996.
- [2] B. Markoski, D. Radosav, P. Vasiljević, Z. Milošević, "Košarkaška statistika", International Conference Dependability and quality management ICDQM 2009, Belgrade, Serbia, pp. 776–781, 2009.
- [3] FIBA, "Official Basketball Rules", 2008.
- [4] FIBA, "Basketball Statisticians' Manual", 2008.
- [5] B. Markoski, Đ. Adžić, "Razlike u vođenju statistike", Trener X, broj 42, Serbia, pp. 24–27, 2006.
- [6] B. Markoski, P. Nemec, Đ. Adžić, "Logičnosti i nelogičnosti u statistici", Trener XII, broj 46–47, Serbia, pp. 19–19, 2008.
- [7] B. Markoski, Đ. Adžić, "Statistika - neki problemi u praksi", Trener XI, broj 46–47, Serbia, pp. 24–28, 2007.
- [8] D. Mihajlović, "Informacioni sistemi i projektovanje baze podataka", Novi Sad, 1998.
- [9] I. Stanojević, D. Surla, "Uvod u objedinjeni jezik modeliranja", Grupa za informacione tehnologije, Novi Sad, 1999.
- [10] H. Borrie, "The Firebird Book: a Reference for Database Developers", 2004.