./

Report – Tennis Scoreboard Tracker

Course Code: <CODE>



Version Number:

Team Members :

Team No:

Module: Model Based System Engineering

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ver. Rel. No.** | **Release Date** | **Prepared. By** | **Reviewed By** | **Approved By** | **Remarks/Revision Details** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**Document History**

# 

**CONTENTS**

1. PROBLEM STATEMENT **4**
2. REQUIREMENTS **5**
   1. Design Constraints **5**
   2. Assumptions **5**
   3. Data Structures **5**
   4. Functional Capabilities **5**
   5. Software Requirements **5**
   6. Detailed Functional Requirements **6**
      1. Umpire Use Case **6**
      2. Public Use Case **6**
   7. Detailed Non Functional Requirements **6**
      1. Security **6**
   8. Requirement Specification Table **7**
3. DESIGN **8**
   1. High Level & Low Level Designs **8**
   2. Point System In Tennis **9**
4. TEST PLAN **11**
   1. Test Strategy **11**
      1. Features To Be Tested **11**
      2. Type Of Testing **11**
   2. Test Objective **11**
   3. Test Criteria **11**
      1. Suspension Criteria **11**
      2. Exit Criteria **11**
   4. Test Environment **11**
5. TEST CASES **12**
6. EXPECTED RESULT **15**
7. CONCLUSIONS **16**
8. REFERENCES **17**
9. **PROBLEM STATEMENT**

It is quite difficult for an official chair umpire to keep track of all the details of an ongoing tennis match in pen and paper. So to make this process much more efficient and easy, we have a robust c software project developed which allows the umpire to simply input the points during real time to get the live score of the ongoing match. Along with the match score, he can also give additional details of the match and store it as files in hard drives. There are also provisions that allow the public in general (including the umpire) to view any previous stored match details, which is quite informative.

**REQUIREMENTS**

***2.1 Design Constraints***

* Should provide details of a given record within a show time of less than 1 second.
* View updated score after each point entry.

***2.2 Assumptions***

* Challenges raised by players are not taken into review for this system.
* Injury forfeits by players not taken into consideration.
* All the umpires are provided with a common login password.

***2.3 Data Structures***

* match - To store details of a match like day of match, tournament etc.
* team - To store details of a team like team name, player name etc.

***2.4 Functional Capabilities***

* menu(); - Displays the menu page which allows to either track a live match or view details of an already completed match.
* scoreboard(); - Computes points during the match and returns it.
* display\_score(); - Displays the scoreboard on a regular basis during score input.
* print\_points(); - Convert points into displayable fashion according to tennis standards ie, 1 into 15, 2 into 30 and 3 into 40.
* menu\_option(); - A function called to perform the 2 menu operations.
* current\_game\_score(); - A function that selects suitable formats to print the current game scores.

***2.5 Software Requirements***

* C program software implemented in Code::Blocks IDE.
* GNU GCC Compiler.

**2.6 *Detailed Functional Requirements***

**2.6.1 *Umpire Use Case***

Use case: Enter points.

Brief description: A secured login provides only an official chair umpire the privilege to enter the points during an ongoing tennis match.

**2.6.2 *Public Use Case***

Use case: View match details.

Brief description: This is a general case, where anyone can go and check any of the completed match details that were stored into the hard disk.

**2.7 *Detailed Non* *Functional Requirements***

**2.7.1 *Security***

The match details can only be written by an official umpire with user-id and login privileges. There are no restrictions for who can view completed match details. These details are made fully available to the general public.

**2.8 *Requirement Specification Table***

|  |  |
| --- | --- |
| REQUIREMENT ID | DESCRIPTION |
| SW\_101\_HP | User can navigate between different windows |
| SW\_102\_HP | Chair umpire can record points |
| SW\_103\_HP | Authentication of user who is trying to record match details |
| SW\_104\_HP | Points should be updated after each point entry |
| SW\_105\_LP | Should comply to ITF(International Tennis Federation) rules and regulations |
| SW\_106\_HP | Points should be displayed after every update |
| SW\_107\_HP | After completion of game, user should be redirected to the menu window |
| SW\_108\_LP | Background colour for the match tracker window should be blue of value 3 |
| SW\_109\_LP | Background colour for the match review window should be grey of value 8 |
| SW\_110\_HP | User is able to exit from the menu screen |
| SW\_201\_LP | Return match details in less than 1 second |
| SW\_202\_HP | All of the match details should be displayed in the match review window |

**TABLE1: Requirement Specification Table**

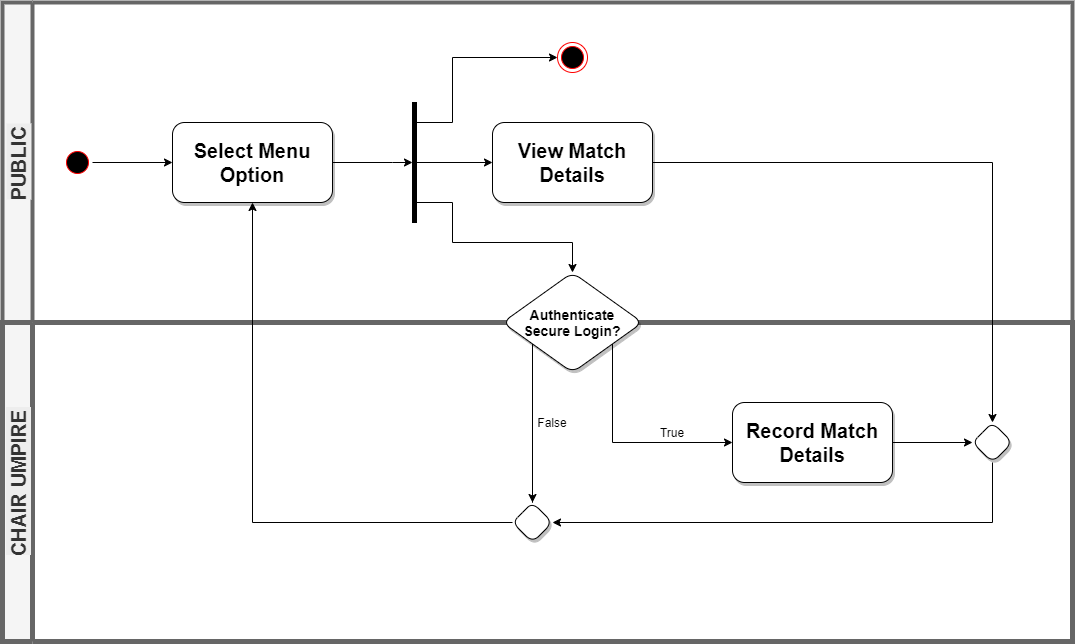
SW : SOFTWARE

LP : LOW PRIORITY

HP : HIGH PRIORITY

1. **DESIGN**

***3.1 HIGH LEVEL & LOW LEVEL DESIGNS***



**FIG 1: Activity Diagram**

The detailed activity diagram shows how the program flow happens. There are basically four window pages available for a user to navigate through, namely:

1. Menu window
2. Login window
3. Match track window
4. Match review window

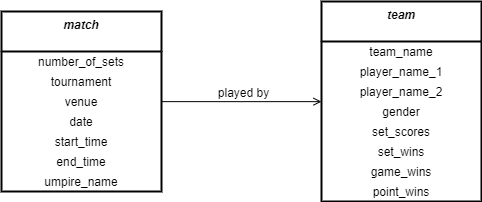
In the menu window, there are three options provided, to track a live match, to view the match details or else to exit from the program.

Now if the user (public) selects option 1, he/she is directed to the login window. This window is designed in such a way that only an official chair umpire who has the authentication credentials is allowed to track a live tennis match. If the authentication fails, user is directed to the menu page else he/she can proceed to the match track window.

In the match track window, the umpire can enter match details and start tracking the match. After adjudging the winner, the umpire is redirected back to the menu window.

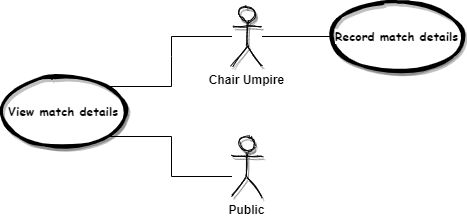
If the public user chooses option 2 from the menu window, he/she can access any completed match file and view the details of that match from the match review window. Afterwards they are redirected to the menu window.

User can exit the software by pressing option 3 from the menu window.



**FIG 2: Class Diagram**

The above class diagram shows the relation between the two structures implemented in the software. The use case diagram below shows different actors that come into play in this software model.



**FIG 3: Use Case Diagram**

***3.2 Point System in Tennis***

* A tennis match is composed of points, games, and sets.
* A set consists of a number of games (a minimum of six), which in turn each consist of points. A set is won by the first side to win 6 games, with a margin of at least 2 games over the other side (e.g. 6–3 or 7–5).
* A match is won when a player or a doubles team has won the majority of the prescribed number of sets. Matches employ either a best of three (first to two sets wins) or best of five (first to three sets wins) set format.
* A ***good serve*** occurs when a legally delivered ball lands in the cross-court service box or on any line bounding it without touching anything in flight.
* A ***let/net*** occurs when a legally delivered ball lands in the cross-court service box having touched the net cord (but not the net post or any other object).
* If the ball fails to clear the net, or bounces anywhere other than the cross-court service box, it is a ***fault***. This is the most common cause of a fault.
* After a fault, play stops immediately. If there has been only one fault on this point, the server is then allowed another attempt.
* If there have been two faults on this point, the point is awarded to the receiver: this is known as a ***double fault***.

1. **TEST PLAN**

***4.1 Test Strategy***

**4.1.1 Feature to be tested**

|  |  |  |
| --- | --- | --- |
| **Function** | **User** | **Description** |
| Track match points | Chair Umpire | A module intended for a chair umpire to update the match points |
| View match points | Public | Module to view details of an already completed match. |

**TABLE2: Features to test**

**4.1.2 Type of Testing**

* Unit testing – Test individual functions. Most our project’s error detection might take place here.
* Integrated testing – Testing of different functions altogether.

***4.2 Test Objective***

The test objectives are to verify the functionality of the tennis scoreboard tracker project, where the testing should be focussed on point tracking system, records to be displayed etc. so as to guarantee that all these operation can work normally in real business environment.

***4.3 Test Criteria***

**4.3.1 Suspension criteria**

Ongoing tests are dropped and an alert is brought to the developer’s attention if more than 20% of the test cases fail.

**4.3.2 Exit criteria**

* Run rate is required to be 100%.
* Pass rate has to be equal to or more than 95%.

***4.4 Test Environment***

The testing is done using the Unity tool in Code::Blocks environment.

1. **TEST CASES**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Case ID | Test Scenario | Test Steps | Test Data | Expected Result | Actual Result |
| T01 | Check point to print on display screen | 1. Pass point to function | Value = 1 | “15” | 15 |
| T02 | Check point to print on display screen | 1. Pass point to function | Value = 2 | “30” | 40 |
| T03 | Check point to print on display screen | 1. Pass point to function | Value = 3 | “40” | 40 |
| T04 | Check points are updated after each entry by umpire | 1. Enter team A has won a point | A/a | Update in team A’s point | Updated |
| T05 | Check chair umpire login credentials | 1. Enter umpire name 2. Enter the correct password | Name - any name  Password-“notaseasyasitseems” | Able to login | Able to login |
| T06 | Check proper exit via menu | 1. Press option for exit from menu window | 3 | Proper exiting | Proper exiting |
| T07 | Check background colour on match tracker window | 1. Option 1 in menu window 2. Authenticate as umpire | - | Blue  (Value = 3) | Background colour of value 3. |
| T08 | Check background colour on match review window | 1. Option 2 in menu window | - | Grey  (Value = 8) | Background colour of value 8. |
| T09 | Check if umpire is redirected back to menu window | 1. Press enter after completion of game | “\n” | Able to return to menu window | Returns to menu window |
| T10 | Check if net serves were accounted | 1. Input net serve in the match tracker window | N/n | “Net Serve. Serve Again” | Net Serve. Serve Again |
| T11 | Check chair umpire login with invalid credentials | 1. Enter umpire name 2. Enter the incorrect password | Name - any name  Password – “veryeasy” | Unable to login | Unable to login |
| T12 | Check points are updated after each entry by umpire | 1. Enter team B has won a point | B/b | Update in team B’s point | Updated |
| T13 | Check points are not updated after each invalid entry by umpire | 1. Enter team A has won a point | 15 | No change in point | No update |
| T14 | Check points are not updated after each invalid entry by umpire | 1. Enter team A has won a point | +1 | No change in point | No update |
| T15 | Check navigation from menu window to match review window | 1. Select option to view match details from menu window | 2 | Menu review window opens | Menu review window has opened |
| T16 | Check navigation from menu window to login window | 1. Select option to track a match from menu window | 1 | Login window opens asking authentication | Login window has opened |
| T17 | Check if fault serves were accounted | 1. Input a fault serve in match tracker window | F/f | “Fault. Serve Again.” | Fault. Serve Again.” |
| T18 | Check if double fault results in a point update | 1. Input 2 consecutive faults in match tracker window | F/f + F/f | “Double Fault. Point Receiver.” | Double Fault. Point Receiver |

**TABLE3: Test Cases**

1. **EXPECTED RESULT**

* All details of a match are tracked onto a file record and points are displayed after every successive entry by the official chair umpire.
* Completed match details are successfully displayed including the set scores.
* Successful exit by the user from menu window.

1. **CONCLUSION**

A detailed study was conducted on the tennis scoreboard tracker software implemented using C. Requirements and designs were planned ahead according to standard tennis rules and test plans were drawn. The project will relieve the chair umpires and other officials from the challenge of taking note of each game point and instead, allow them to easily handle the point calculations during a live ongoing match. The project also will allow people to gather any information about a match by searching through the directory at easy and within a blink of an eye.

For future development, the software can further developed into a secure application on a mobile/tab for chair umpires and after completion can upload the match details directly onto the official ITF website’s database, making it a clear and transparent move of information.

1. **REFERENCES**
2. <https://en.wikipedia.org/wiki/Tennis_scoring_system>
3. IEEE Guide for Software Requirements Specifications," in *IEEE Std 830-1984* , vol., no., pp.1-26, 10 Feb. 1984, doi: 10.1109/IEEESTD.1984.119205.
4. https://www.itftennis.com/media/2510/2020-rules-of-tennis-english.pdf