Sanjay Ghodawat University Kolhapur

Established as a State Private University under Govt. of Maharashtra Act No. XL dated 3rd May 2017

Empowering Lives Globally !

PROJECT REPORT

A report submitted in partial fulfillment of the requirements for the

Project on

"Cricket Scoreboard Sheet"

School of Computer Science & Engineering

By

Sankalp S. Misal
22SC114281057 50

Tejas S. Dhole
22SC114281058 52

Mangesh S.Patil
22SC114281066 58

Prajwal C. Kamate
22SC114281047 43

Program: Bachelor of Computer Science and Engineering Class: FY BTech

Under Supervision of

Mrs. Veena Mali

Academic Year: 2022-2023

Sanjay Ghodawat University Kolhapur

Established as a State Private University under Govt. of Maharashtra Act No. XL dated 3rd May 2017

Empowering Lives Globally !

School of Computer Science & Engineering

CERTIFICATE

This is to certify that the "**Project Report**"

On

"Cricket Scoreboard Sheet"

submitted by

Sankalp S. Misal
22SC114281057 50

Tejas S. Dhole
22SC114281058 52

Mangesh S.Patil
22SC114281066 58

Prajwal C. kamate
22SC114281047 43

Program: B Tech Class: FY B Tech (Div A)

is work done by him/her and submitted during the 2022 – 2023 academic year, in partial fulfillment of the **Project.**

Sanjay Ghodawat University, Kolhapur

Mrs. Veena Mali	**Mrs. Deepika Patil**	**Dr. B. Suresh Kumar**	**Examiner**
Project Guide	**PBL Co-ordinator**	**Head, SOCSE**	

DECLARATION

I the undersigned solemnly declare that the report of the project work entitled **"Cricket Scoreboard Sheet"** which is carried out under the supervision of **Mrs. Veena Mali.** I assert that the statements made and conclusions drawn are an outcome of the project work. I further declare that to the best of my knowledge and belief that the project report does not contain any part of any work which has been submitted for the award of any other degree/diploma/certificate in this University or any other University.

Student Name:

Sankalp S. Misal
22SC114281057 50

Tejas S. Dhole
22SC114281058 52

Mangesh S.Patil
22SC114281066 58

Prajwal C. kamate
22SC114281047 43

Class: FY B Tech (Div A)

ACKNOWLEDGMENT

First, I would like to thank my Head of the School **Dr. B. Suresh Kumar** for constructive criticism throughout my project. I would like to thank PBL coordinator **Mrs. Veena Mali** and Department Project Guide **Mrs. Veena Mali** for support and advices to get and complete internship in above said organization. It is indeed with a great sense of pleasure and immense sense of gratitude that I acknowledge the help of these individuals. I am extremely grateful to my department staff members and friends who helped me in successful completion of this project.

ABSTRACT

The project begins by displaying the welcome screen, which fades up to reveal the main menu. There are three choices on the main menu:

- Create a new score sheet
- View a previous score sheet
- Exit

If '1' is entered, the Cricket Score Sheet project prompts for a new score sheet's name. A notice appears on the screen when the file is generated.
The user must next fill out the score sheet, which includes the following information:

1. Competition
2. Venue
3. Match between and versus
4. Toss winner team
5. Elected choice of toss winner
6. Inning and date
7. Name of batsman and run hit by each of them
8. Name of bowler and run given by each blower

After entering these details, the application prompts the user to press 'e' to amend the information and 'c' to proceed.
When the user selects '2' from the main menu, the application prompts for the file name. The file is shown if it is found. Otherwise, the screen displays an error message.
Exit is the third option on the main menu. The Cricket Score Sheet project is terminated if the number '3' is entered in the main menu.

TABLE OF CONTENT'S

Introduction

The project cricket scorecard developed in c is used to provide user with an update of the cricket even when the user is not watching the mach. The user can use this website anytime, anywhere to see the teams, matches, player's squad, runs scored by each player and can also view the reviews and commentary. This gives original experience of watching the match by the user.

Software or website developed must be built from user's point of view. It must be able to fulfil all the drawbacks that user's face in existing system. Our system fulfils and satisfies the user also it gives the experience of watching the match by adding the commentary in the website. The website is maintained by admin where he/she updates the score without any delay.

Adding teams, matches, players and score of each player is the responsibilities of admin. User can just register with their basic details and login to the website to view the team details and scores of individual.

The system overcomes all the drawbacks and maintains the website up to date.

- **Problem Definition:**

1. **Time consuming:** The manual processing is taking more time. It takes lots of time to record the process and transaction into a paper.

2. **Security is not assured:** Security is not assured for the records of the organization. The need for computerizing arises in order assure the security of the records from fire or other destruction.

3. **Space consuming:** A lot of space is required to maintain the record physically. To solve the problem they are going for computerization.

- **Scope:**

The scope of a cricket score board sheet typically includes the recording and display of various important information and statistics related to a cricket match. Here are some common elements you would find on a cricket score board sheet:

1.Team Names: The names of the teams participating in the match are usually displayed at the top of the score board.

2.Batting Team's Score: The score of the batting team is displayed prominently. It shows the total number of runs scored by the team and the number of wickets lost.

3.Batsmen Details: The score board includes information about the batsmen currently at the crease. It includes their individual scores, the number of balls faced, and the dismissal method if they get out.

4.Bowler Details: The score board provides information about the bowlers who are currently bowling. It includes the number of overs, maidens, runs conceded, and wickets taken by each bowler.

5.Extras: The score board shows the number of extra runs conceded by the fielding team, such as wides, no balls, and byes.

6.Run Rate: The run rate indicates the average number of runs scored per over by the batting team. It is calculated based on the current score and the number of overs bowled.

7.Fall of Wickets: The score board records the sequence in which the batsmen got out, including the score at which each wicket fell.

8.Partnership: The score board displays the partnership details between two batsmen, including the number of runs scored in their partnership.

9.Match Information: The score board provides information about the match, such as the venue, date, and session of play (e.g., morning, afternoon, or evening).

Problem Identification

1. Manual Data Entry Errors: One of the primary issues with a cricket score board sheet is the potential for human error during the manual recording of scores, wickets, runs, and other statistics. This can result in incorrect information being displayed, leading to inaccurate analysis and judgments.

2.Time-consuming Process: Maintaining a physical score board sheet requires continuous attention and manual updates after every ball, over, or event. This process can be time-consuming, especially for high-intensity matches or tournaments, and may delay the availability of real-time information.

3.Limited Accessibility: The physical score board sheet is typically available only at the cricket ground, limiting its accessibility to those who are physically present. This can be inconvenient for fans, media personnel, and individuals who are unable to attend the match but still want to stay updated.

4.Lack of Real-time Updates: Unless the score board sheet is being updated digitally or broadcasted on television, the information may not be available in real time to a wider audience. This can hinder the ability of cricket enthusiasts to follow the match or make informed decisions based on the current state of play.

5.Incomplete Statistical Analysis: While a score board sheet captures the basic statistics of a cricket match, it may not provide comprehensive analysis or advanced metrics that can aid in evaluating player performance, team strategies, or match trends. This limitation can hinder in-depth analysis and understanding of the game.

6.Dependency on Weather Conditions: Physical score board sheets are susceptible to weather conditions such as rain, wind, or extreme temperatures. They can get wet, damaged, or even blown away, resulting in loss of data and disruptions during the match.

Objectives

1. It uses file management to store data such as runs, wickets, overs and extras among others.
2. The program can display runs, wickets, batsman and bowler names, overs, bonuses, bowler's economy, batsman's strike rate and other statistics.
3. The source code is thorough, error-free and easy to understand.
4. Easily maintain all the player details.
5. Report generation is easier
6. Easy to maintain score details.
7. Ensure user security.

System Requirements Specification

- **Software Requirement:**

  - Turbo C
  - Microsoft Visual Studio Code
  - Dev C++

- **Hardware Requirement**

  - Computer or laptop
  - Intel(R) Core(TM) i3-Processor
  - RAM-1 GB Minimum
  - Storage-10

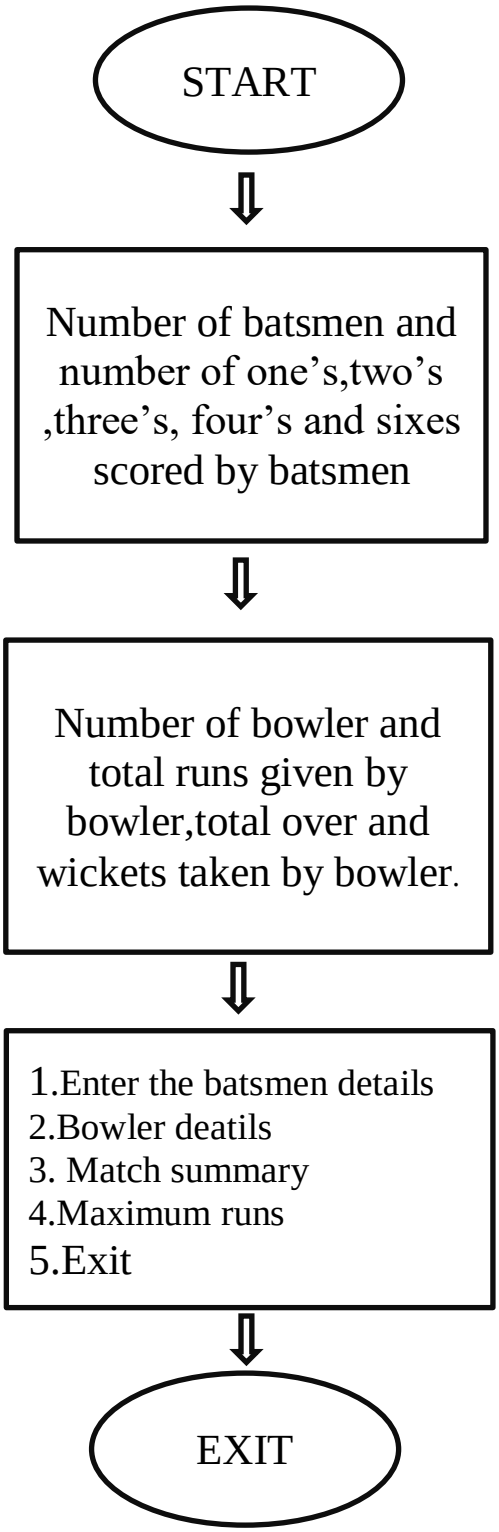
Methodology

The project entitled with "Cricket Score Card System" is divided into numerous modules. The detail description about the whole modules will be explained in below.

Fig 1 depicts the detail functionality of the Cricket Score Card System with the connection of the modules like Admin, User, Commentary and Review. Admin The admin will update details of upcoming cricket matches, create and block user accounts.

The admin module is the major module as it is responsible for carrying out the major operations regarding site updates, score updates etc., It maintains information regarding other modules. The various software components in administrator module updates the information about match details, player details. Censoring of comments can be done during or after the match by the admin.

- **Flow Diagram(Flow Chart):**



Implementation

1.Implementing the score mechanics: The first objective is to implement the score mechanics, adding player details,dispalys score of each individual,user can use website any time.

2.Creating the game environment: The second objective is to create the game environment, it is easy to use at any place anywhere.

3.Handling user input: The third objective is to handle user input,if we input player name it automatically process and program executes.

4.Implementing game logic: The fourth objective is to implement game logic, it will br more difficult to main player details individually.It will take more time

“Introduction of C Programming”

The native language of a computer is binary—ones and zeros—and all instructions and data must be provided to it in this form. Native binary code is called machine language. The earliest digital electronic computers were programmed directly in binary, typically via punched cards, plug-boards, or front-panel switches. Later, with the advent of terminals with keyboards and monitors, such programs were written as sequences of hexadecimal numbers, where each hexadecimal digit represents a four binary digit sequence. Developing correct programs in machine language is tedious and complex, and practical only for very small programs.

In order to express operations more abstractly, assembly languages were developed. These languages have simple mnemonic instructions that directly map to a sequence of machine language operations For example, the MOV instruction moves data into a register, the ADD instruction adds the contents of two registers together. Programs written in assembly language are translated to machine code using an assembler program. While assembly languages are a considerable improvement on raw binary, they still very low-level and unsuited to large-scale programming. Furthermore, since each processor provides its own assembler dialect, assembly language programs tend to be non-portable; a program must be rewritten to run on a different machine.

The 1950s and 60s saw the introduction of high-level languages, such as Fortran and Algol. These languages provide mechanisms, such as subroutines and conditional looping constructs, which greatly enhance the structure of a program, making it easier to express the progression of instruction execution; that is, easier to visualise program flow

Advantages of learning C:

- C is a universal language that can be used for various applications.
- C is a very efficient language that can write code that is both fast and reliable.
- C is a portable language, meaning that code written in C can be easily compiled and run on various platforms.

Source Code:

```
#include <stdio.h>
#include <stdlib.h>
struct batsman
{
   char name[25];
   int runs, score, balls, toruns, toball, ones, twos, threes, fours, sixes;
   int max_six, max_run, max_four;
   float str;
} pl1[100], pl3;
struct bowler
{
   char name[25];
   int runsgv, wkttkn, overs;
   int max_w;
   float econ;
} pl2[100], pl4;
int main()
{
   int plno, choice;
   int i, n, m;
   printf("Enter the Batsman detail:\n");
   printf("Enter the number of batsman:\n");
   scanf("%d", &m);
   for (i = 0; i < m; i++)
   {
      printf("Enter name of batsman%d:\n", i + 1);
      scanf("%s", pl1[i].name);
      printf("Enter the number of ones scored by player%d:\n ", i + 1);
      scanf("%d", &pl1[i].ones);
      printf("Enter the number of twos scored by player%d:\n ", i + 1);
      scanf("%d", &pl1[i].twos);
      printf("Enter the number of threes scored by player%d:\n ", i + 1);
      scanf("%d", &pl1[i].threes);
      printf("Enter the number of fours scored by player%d:\n ", i + 1);
      scanf("%d", &pl1[i].fours);
      printf("Enter the number of sixes scored by player%d:\n ", i + 1);
      scanf("%d", &pl1[i].sixes);

      printf("Enter the balls played by the player%d:\n", i + 1);
      scanf("%d", &pl1[i].balls);
   }

   printf("\nEnter the bowlers details:\n");

   printf("Enter the number of bowlers:\n");

   scanf("%d", &n);

   for (i = 0; i < n; i++)
```

```
    {

        printf("\nEnter name of bowler%d:", i + 1);
        scanf("%s", pl2[i].name);

        printf("Enter the runs given by the bowler%d:\n ", i + 1);
        scanf("%d", &pl2[i].runsgv);

        printf("Enter the overs bowled by the bowler%d:\n", i + 1);
        scanf("%d", &pl2[i].overs);

        printf("Enter the wickets taken by the bowler%d\n", i + 1);
        scanf("%d", &pl2[i].wkttkn);
    }

    printf("Thank you all details are recorded\n");

    do
    {

        printf("Enter the choice:\n 1)Batsman detail:\n 2)Bowlers detail:\n 3)Match summary:\n
4)Record:\n 5)Exit\n ");
        scanf("%d", &choice);

        switch (choice)
        {
        case 1:
            printf("Enter the batsman number to see his details\n");
            scanf("%d", &plno);

            plno--;
            printf("                    Player Detail\n");
            printf("= = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = =
= = = =\n");
            printf(" Batsman       runs          balls       fours       sixes        sr   \n");
            printf("= = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = =
= = = =\n");
            pl1[plno].runs = (1 * pl1[plno].ones) + (2 * pl1[plno].twos) + (3 * pl1[plno].threes) + (4 *
pl1[plno].fours) + (6 * pl1[plno].sixes);
            pl1[plno].str = (pl1[plno].runs * 100.00) / pl1[plno].balls;
            printf(" %-15s %-14d %-13d %-11d %-11d %-9.2f\n\n", pl1[plno].name, pl1[plno].runs,
pl1[plno].balls, pl1[plno].fours, pl1[plno].sixes, pl1[plno].str);

            break;

        case 2:
            printf("Enter the bowlers number to see his details\n");
            scanf("%d", &plno);

            plno--;
            printf("                    Player Detail\n  ");
printf(" = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = =
```

```
= = = = = = \n");
          printf(" Bowler        overs           runs        wicket       economy\n");
          printf(" = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = =
= = = = = = \n");
          for (i = 0; i < n; i++)
          {
             pl2[plno].econ = pl2[plno].runsgv / pl2[plno].overs;
             printf(" %-15s %-14d %-13d %-11d %-11.2f\n\n", pl2[plno].name, pl2[plno].overs,
pl2[plno].runsgv, pl2[plno].wkttkn, pl2[plno].econ);
          }

          break;

       case 3:
          printf("                    Match summary\n");
          printf(" = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = =
= = = = = = = = = = = = = = = = = = \n");
          printf(" Batsman        runs          balls        fours       sixes        sr   \n");
          printf(" = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = =
= = = = = = = = = = = = = = = = = = \n");
          for (i = 0; i < 1; i++)
          {
             pl1[i].runs = (1 * pl1[i].ones) + (2 * pl1[i].twos) + (3 * pl1[i].threes) + (4 * pl1[i].fours) +
(6 * pl1[i].sixes);
             pl3.toruns += pl1[i].runs;
             pl1[i].str = (pl1[i].runs * 100.00) / pl1[i].balls;
             printf(" %-15s %-14d %-13d %-11d %-11d %-9.2f\n\n", pl1[i].name, pl1[i].runs,
pl1[i].balls, pl1[i].fours, pl1[i].sixes, pl1[i].str);
          }
          printf("TOTAL RUNS:%d\n\n", pl3.toruns);
          printf("\n\n");
          printf("= = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = =  = = =
= = = = = =  = = = = = \n");
          printf(" Bowler        overs           runs        wicket       economy\n");
          printf(" = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = =
= = = = = = = = = = = =\n");
          for (i = 0; i < n; i++)
          {
             pl2[i].econ = pl2[i].runsgv / pl2[i].overs;
             printf(" %-15s %-14d %-13d %-11d %-11.2f\n\n\n", pl2[i].name, pl2[i].overs,
pl2[i].runsgv, pl2[i].wkttkn, pl2[i].econ);
          }

          break;

       case 4:
          pl3.max_run = 0, pl4.max_w = 0, pl3.max_four = 0, pl3.max_six = 0;

          for (i = 0; i < m; i++)
          {
             pl1[i].runs = (1 * pl1[i].ones) + (2 * pl1[i].twos) + (3 * pl1[i].threes) + (4 * pl1[i].fours) +
(6 * pl1[i].sixes);
```

```
            if (pl3.max_run < pl1[i].runs)
            {
               pl3.max_run = pl1[i].runs;
            }

            if (pl3.max_six < pl1[i].sixes)
            {
               pl3.max_six = pl1[i].sixes;
            }

            if (pl3.max_four < pl1[i].fours)
            {
               pl3.max_four = pl1[i].fours;
            }

            if (pl4.max_w < pl2[i].wkttkn)
            {
               pl4.max_w = pl2[i].wkttkn;
            }
         }
         printf("Highest runs scored by the batsman:%d\n", pl3.max_run);

         printf("Maximum fours scored by the batsman:%d\n", pl3.max_four);

         printf("Maximum sixes scored by the batsman%d:\n", pl3.max_six);

         printf("Maximum wickets taken by the bowler:%d\n", pl4.max_w);

         break;

      case 5:
         exit(1);

      default:
         printf("Enter the correct choice\n");
         break;
      }

   } while (choice != 5);

   return 0;
}
```

Result

```
Enter the Batsman detail:
Enter the number of batsman:
 4
Enter name of batsman1:
Rohit
Enter the number of ones scored by player1:
 1
Enter the number of twos scored by player1:
 2
Enter the number of threes scored by player1:
 3
Enter the number of fours scored by player1:
 4
Enter the number of sixes scored by player1:
 6
Enter the balls played by the player1:
50
Enter name of batsman2:
Virat
Enter the number of ones scored by player2:
 1
Enter the number of twos scored by player2:
 4
Enter the number of threes scored by player2:
 2
Enter the number of fours scored by player2:
 2
Enter the number of sixes scored by player2:
 1
Enter the balls played by the player2:
33
Enter name of batsman3:
Hardik
Enter the number of ones scored by player3:
 4
Enter the number of twos scored by player3:
 1
Enter the number of threes scored by player3:
 1
Enter the number of fours scored by player3:
 1
Enter the number of sixes scored by player3:
 1
Enter the balls played by the player3:
35
Enter name of batsman4:
Shikhar
Enter the number of ones scored by player4:
 1
Enter the number of twos scored by player4:
 4
```

Enter the number of threes scored by player4:
 2
Enter the number of fours scored by player4:
 2
Enter the number of sixes scored by player4:
 2
Enter the balls played by the player4:
24

Enter the bowlers details:
Enter the number of bowlers:
2

Enter name of bowler1:Kunal
Enter the runs given by the bowler1:
 23
Enter the overs bowled by the bowler1:
3
Enter the wickets taken by the bowler1
0

Enter name of bowler2:Jadeja
Enter the runs given by the bowler2:
 43
Enter the overs bowled by the bowler2:
20
Enter the wickets taken by the bowler2
0

Thank you all details are recorded
Enter the choice:
 1)Batsman detail:
 2)Bowlers detail:
 3)Match summary:
 4)Record:
 5)Exit

Enter the batsman number to see his details
1

Player Detail

= =

Batsman	runs	balls	fours	sixes	sr
Rohit	66	50	4	6	132.00

Enter the choice:
 1)Batsman detail:
 2)Bowlers detail:
 3)Match summary:
 4)Record:
 5)Exit

Enter the batsman number to see his details
2

Player Detail

Batsman	runs	balls	fours	sixes	sr
Virat	29	33	2	1	87.88

Enter the choice:
1)Batsman detail:
2)Bowlers detail:
3)Match summary:
4)Record:
5)Exit

Match summary

Batsman	runs	balls	fours	sixes	sr
Rohit	66	50	4	6	132.00

TOTAL RUNS:66

Bowler	overs	runs	wicket	economy
Kunal	3	23	0	7.00
Jadeja	20	43	0	2.00

Highest runs scored by the batsman:66
Maximum fours scored by the batsman:4
Maximum sixes scored by the batsman6:
Maximum wickets taken by the bowler:0

Conclusion & Future Scope

- **Future Scope:**

In the future the scope of a cricket scoreboard program written in c language can be expanded in various ways to incorporate in more advanced features.

1. User interface in enhancement: develop a graphical user interface(GUI) using libraries.
2. It also provides a more visually appealing and user friendly scoreboard display.
3. Live score updates.
4. We can also develop a cricket fantasy application using for scoreboard presentation.

- **Conclusion:**

It is concluded that the application works well and satisfy the end users. The application is tested very well and errors are properly debugged. The application is simultaneously accessed from more than one system. Simultaneous login from more than one place is tested. This system is user friendly so everyone can use easily. Proper documentation is provided. The end user can easily understand how the whole system is implemented by going through the documentation. The system is tested, implemented and the performance is found to be satisfactory. All necessary output is generated. Thus, the project is completed successfully. Further enhancements can be made to the application, so that the application functions very attractive and useful manner than the present one. The speed of the transactions become more enough now.

References

Websites:

1. Studytonight:- https://www.studytonight.com/c-projects/cricket-score-display-project-using-c-language

2. javatpoint:- https://www.javatpoint.com/cricket-score-sheet-in-c

Books:

1. Let us C ~ Yashavant Kanetkar