**POKHARA UNIVERSITY**

**CRIMSON COLLEGE OF TECHNOLOGY**

**Devinagar-11, Butwal**



**A First Year Project Report On**

**CRICKET SCORE BOARD.**

*(PRJ– 151)*

*A project report submitted on behalf of partial fulfillment of Bachelor of Computer Application*

**Pokhara University, Nepal**

**Submitted To:**

**Crimson College of Technology**

Department of Science and Technology

**Submitted By:**

Ankit Bashyal (Symbol no:23530224) Prince Kharal (Symbol no:23530246)

Nirajan Pandey (Symbol no:23530240) Ayush Khanal (Symbol no: 23530231)

*September, 2024*

**SUPERVISOR’S RECOMMENDATION**

I hereby recommend that this report has been prepared under my supervision by **Ankit Bashyal** (Symbol no: 23530224), **Nirajan Pandey** (Symbol no:23530240), **Prince Kharal** (Symbol no: 23530246), **Ayush Khanal** (Symbol no: 23530231) entitled **“CRICKET SCORE BOARD ”** in partial fulfillment of the requirements for the degree of BCA (Bachelor of Computer Application) be processed for evaluation.

……..……................................................

**Mr. Nabaraj Koirala**

**Project Supervisor**

**Department of Science and Technology**

**Crimson College of Technology**

**CERTIFICATE FROM EXTERNAL EXAMINER**

This is to certify that the work carried out by **“** **Ankit Bashyal (23530224), Nirajan Pandey (23530240), Prince Kharal (23530246) and Ayush Khanal (23530231)”** for the completion of the project entitled **CRICKET SCORE BOARD** in the fulfilment of the requirement of the degree of Bachelor of Computer Application(BCA), has been accomplished, presented, and demonstrated successfully. During project, I have found the students to be enthusiastic, hardworking, punctual and ready-with-skills to undertake any work related IT .

……………………………………….

EXTERNAL EXAMINER

……………………………………….

……………………………………….

**PREFACE**

Sports and technology are getting more and more entwined in today's society, as digital solutions improve the administration and presentation of different sports. The popular game of cricket is not an exception worldwide. The Cricket Scoreboard is a crucial component of the game of cricket since it provides real-time tracking of individual performance, team statistics, and match progress.

The goal of this project, which goes by the name Cricket Scoreboard, is to automate the process of gathering and presenting statistics about cricket matches. The project, which was created as a requirement for the second semester of the Bachelor in Computer Applications (BCA) program, focuses on how to combine structured data management with programming techniques to produce a system that is easy to use. The project's main objective is to imitate a digital scoreboard for cricket, where users can enter, compute, and show player and match information.

The project not only shows how basic C programming concepts, including data structures, file management, and algorithms, can be applied, but it also highlights how manual work may be made simpler by using contemporary software. Our goal with this project is to provide a desktop program that is easy to use, efficient in handling cricket score data, and scalable.

The project not only shows how basic C programming concepts, including data structures, file management, and algorithms, can be applied, but it also highlights how manual work may be made simpler by using contemporary software. Our goal with this project is to provide a desktop program that is easy to use, efficient in handling cricket score data, and scalable.

**ACKNOWLEDGEMENT**

In the completion of this project entitled **"CRICKET SCORE BOARD "** all the members of the group has toiled equally. Foremost we would like to thank **Crimson College of Technology** for assigning this type of project that would help in our future carrier development. We are also thankful to **Mr. Nabaraj Koirala**, Project supervisor who co-operated and provided information about the project and without whose supervision or would not have been possible to get up to this level of project.

We would like to express our sincere thanks to **Mr. Abdul Hak**, IT coordinator for his help and interest in our project and also for providing us his valuable suggestion for the improvement of the project.

Last but not the least; we would like to express our gratitude towards all our teachers, seniors, and friends for their lovely support and guidance.

Thanking from...

……………………

Mr. Ankit Bashyal

……..……................................................

Mr. Abdul Hak

Project Supervisor

Department of Science and Technology

Crimson College of Technology

……………………

Mr. Nirajan Pandey

……………………

Mr. Ayush Khanal

…………………...

Mr. Prince Kharal

**ABSTRACT**

The proposal entitled **‘CRICKET SCORE BOARD’** is prepared as partial requirement for the completion of two credit hour course of **Bachelor of Computer Application** as the second semester project.

The use of computer has made the world smaller. It is quite impossible to think of any information system without the implementation of the computer technology. Computer technology has affected the entire field such as business field, scientific field, management field, etc. Users like to access for different information of the Employee and their records. The project mainly emphasizes on providing the information related to the Employee.

The report presents the detail view of the project. At present time, it has become very essential to manage the information related to the customers. Our project helps the user to manage and display player statistics in a cricket match. In summary, this project simulates a basic cricket scoreboard system, allowing users to input, calculate, and display player statistics and match summaries in a user-friendly way.

Our project is based on the experimental research. The most common and standard programming language Visual Basic is used and the project is based on the general purpose language to develop a window platform. Since, many window database programs and other application package is written is Visual Basic, SQL and MS-Access. The process of data modeling has simplified the implementation of database through the use of data flow diagram and use-case diagram.

This desktop application provides easy way to manage the records. This application is designed in such a way that any further enhancement can be done with ease. It is developed by using Vb. Net for front- end and SQL for back- end. This application is going to be used for any cricket matches to easily manage the match information. So, we are happy to develop this desktop application as it provides all the requirements for the cricket match.

****

TABLE OF CONTENTS

**SUPERVISIOR RECOMMENDTAION ……………………….…………………**

**CERTIFICATE FROM EXTERNAL EXAMINER ……………………………...**

**PREFACE……………………………………………………………………………**

**ACKNOWLEDGEMENT …………………………………………….……………**

**ABSTRACT …………………….………………………...…………………………**

**TABLE OF CONTENT……………………………………………………………...**

**LIST OF FIGURES …………………………………………………………………*i***

**LIST OF TABLES …………………………………………………………………. *ii***

**LIST OF ABBREVIATIONS ……………………………………………………… *iii***

**UNIT 1: INTRODUCTION TO PROJECT**

* 1. [Introduction………………………………………………………………1](#_bookmark8)
  2. [Problem Statement………………………………………………………..2](#_bookmark9)

[1.3. Objectives………………………………………………………………...3](#_bookmark10)

* 1. [Scope and Limitations…………………………………………………….3-4](#_bookmark11)
     1. [Scope…………………………………………………………………..3](#_bookmark12)
     2. [Limitations……………………………………………………………..4](#_bookmark13)

[1.5 Methodology………………………………………………………………5-6](#_bookmark14)

1.6 Report Organization ……………………………………………………... 6

**UNIT 2 BACKGROUND STUDY AND LITERATURE REVIEW**

[2.1 Background Study…………………………………………………………7](#_bookmark18)

[2.2 Literature Review………………………………………………………….8](#_bookmark19)

[2.2.1 Study of existing systems……………………………………………...](#_bookmark20).8

2.2.2 Gaps Identified in Existing Systems in Nepal ……………………………9

**UNIT 3: SYSTEM REQUIREMENT AND ANALYSIS**

**3.1 Hardware Requirement ………………………………………………….10**

**3.2 Software Requirement……………………………………..……………..11**

**3.3 System Analysis and Feasibility Study…………………………………..12-14**

**3.3.1 Technical Feasibility………………………………………………....14**

**3.3.2 Economic Feasibility………………………………………………...14**

**3.3.3 Legal Feasibility……………………………………………………..14**

**3.3.4 Operational Feasibility ……………………………………………...14**

**3.3.5 Schedule Feasibility ………………………………………………..14**

**3.4 Analysis …………………………………………………………………15-18**

**3.4.1 Process Modelling through DFD…………………………………….15-16**

**3.4.2 Data Modelling through ER Diagram …………………………….....17-18**

**UNIT 4: TIME, COST AND TASK DIVISION**

**4.1 Time Division…………………………………………………………….19**

**4.2 Cost Estimation and Division …………………………………………....19**

**4.3 Task Division ……………………………………………………………..20-22**

**UNIT 5: IMPLEMENTATION AND TESTING**

**5.1 Implementation Overview ……………………….……………………….23-24**

**5.2 Testing Overview…………………………………..……………………...25-26**

**5.2.1 Unit Testing …………………………………………………………...25**

**5.2.2 System Testing ………………………………………………………..26**

**UNIT 6: CODE AND SNAPSHOTS**

**6.1 Actual Code..………………………………………………………………27-35**

**6.2 Snapshots…………………………………………………………………..35-40**

**UNIT 7: CONCLUSION AND BIBLIOGRAPHY**

**7.1 Conclusion………………………………………….………………………41**

**7.2 Future Recommendations ………………………….………………………42**

**7.3 Bibliography ……………………………………………………………….43**

***UNIT 1: INTRODUCTION OF THE PROJECT***

****

**1.1: Introduction of Cricket Score Board**

This project is undertaken as part of the Bachelor of Computer Applications (BCA) 2nd semester coursework, with a weightage of 2 credit hours. The primary goal of this project is to apply the programming skills we have learned during the course to a practical, real-world scenario. We chose to develop a **Cricket Scoreboard** software application, as it provides a rich and engaging use case for various core programming concepts, including data structures, algorithms, and input-output operations in C.

**What is a Cricket Scoreboard?**

A **Cricket Scoreboard** is an essential tool used to track the performance of players during a cricket match. It displays real-time statistics for both teams, including runs scored, wickets taken, overs bowled, and individual player performances. In traditional matches, this task is often done manually or using sophisticated electronic scoreboards. However, in our project, we aim to replicate these features through a console-based application using the C programming language.

**1.2 Problem Statements**

In this section, we identify and articulate the key problems that the Cricket Scoreboard system aims to address. Current methods of scoring in cricket, whether manual or through existing digital systems, often face several challenges:

1. **\*\*Manual Errors\*\*:** Traditional manual scoring can lead to inaccuracies due to human error. Mistakes in recording runs, overs, and wickets can affect the integrity of the match statistics.

2. **\*\*Lack of Real-Time Updates\*\*:** Many existing systems do not provide real-time updates, resulting in delays in score reporting. This is particularly problematic during live matches where spectators and players rely on immediate information.

3. **\*\*User Experience\*\*:** Current scoreboard systems can be cumbersome and not user-friendly. Operators may struggle with complicated interfaces, which can lead to frustration and inefficiencies during high-pressure situations.

4. **\*\*Limited Accessibility\*\*:** Some scoring systems may only be available on specific platforms or devices, limiting access for users who wish to follow matches from different locations.

5. **\*\*Inadequate Data Management\*\*:** Existing systems may lack comprehensive features for managing player statistics over time, making it difficult for teams and coaches to analyze performance trends.

6. **\*\*Integration Challenges\*\*:** Many scoring systems do not easily integrate with other cricket-related applications or data sources, hindering the ability to provide comprehensive match analyses or historical data.

By addressing these issues, the proposed Cricket Scoreboard system aims to provide a more accurate, efficient, and user-friendly solution for scoring cricket matches, ultimately enhancing the experience for players, spectators, and officials alike.

**1.3 Objectives**

The primary objectives of the Cricket Scoreboard system are as follows:

**1. \*\*Accuracy\*\*:** To provide a reliable platform for recording and displaying match statistics, minimizing human error through automated calculations for runs, strike rates, and other relevant metrics.

**2. \*\*Real-Time Updates\*\*:** To enable real-time score tracking, allowing players, spectators, and officials to access current match information instantly, thereby enhancing the viewing experience.

**3. \*\*User-Friendly Interface\*\*:** To develop an intuitive and accessible interface that simplifies data entry and retrieval for users of varying technical proficiency, ensuring ease of use during matches.

**4. \*\*Comprehensive Data Management\*\*:** To facilitate the efficient management of player statistics, enabling coaches and teams to analyze performance trends over time for informed decision-making.

**5. \*\*Integration Capabilities\*\*:** To create a system that can easily integrate with other cricket-related applications and databases, allowing for richer data analysis and reporting.

**6. \*\*Enhanced Engagement\*\*:** To improve engagement among players and spectators by providing timely and accurate updates, fostering a more immersive cricket experience.

By achieving these objectives, the system aims to enhance the overall quality and efficiency of cricket scoring.

**1.4 Scope and Limitations**

**1.4.1 Scope**

The Cricket Scoreboard system is designed to serve as a comprehensive tool for managing cricket match statistics. It will allow users to input and store details for both batsmen and bowlers, calculate key performance metrics such as runs, strike rates, and economies, and provide real-time updates on match progress. The system will include features for viewing detailed player statistics and generating match summaries. It aims to support local cricket tournaments, school matches, and community games, providing a user-friendly experience for scorekeepers, players, and fans alike.

#### **1.4.2 Limitations**

While the system aims to address several challenges in cricket scoring, it also has limitations. Firstly, it relies on accurate user input, which can still lead to errors if data is entered incorrectly. The system is designed primarily for local matches and may not incorporate advanced features required for professional cricket, such as ball-by-ball commentary or integration with live broadcast systems. Additionally, its effectiveness is dependent on the availability of suitable hardware and software environments. Lastly, the system may not cover every specific requirement of all cricket formats, such as T20 or Test matches, limiting its adaptability to diverse playing conditions.

### 1.5 Methodology

The development of the Cricket Scoreboard system follows a structured methodology to ensure a comprehensive and effective solution. The approach encompasses several key phases:

1. \*\*Requirement Gathering\*\*: This initial phase involves collecting detailed requirements from potential users, including scorekeepers, players, and coaches. Surveys and interviews help identify the necessary features and functionalities needed to enhance the scoring process.

2. \*\*System Design\*\*: Based on the gathered requirements, a blueprint for the system is created. This includes designing the user interface, defining the database structure for storing player and match data, and outlining the overall architecture of the software.

3. \*\*Implementation\*\*: The actual coding of the system takes place in this phase. The programming is performed using C, as outlined in the code provided. This phase includes developing functions for data entry, calculations, and file management.

4. \*\*Testing\*\*: Comprehensive testing is conducted to identify and rectify any bugs or issues within the system. This includes unit testing individual components and system testing to ensure all features work seamlessly together.

5. \*\*Deployment\*\*: Once testing is completed, the system is deployed for use in real match situations. User feedback is gathered to assess functionality and ease of use.

6. \*\*Maintenance and Updates\*\*: After deployment, ongoing support is provided to address any issues that arise and to implement updates based on user suggestions and technological advancements.

This methodology ensures a systematic approach to development, allowing for a user-centered design that addresses the specific needs of cricket scoring while ensuring reliability and efficiency.

1.6 Report Organization

This report is structured to provide a comprehensive overview of the Cricket Scoreboard system, detailing its development process, functionalities, and findings.

- \*\*Unit 1: Introduction\*\* sets the foundation by outlining the project's purpose, identifying key problems in existing systems, and stating the objectives that guide the development of the Cricket Scoreboard.

- \*\*Unit 2: Background Study and Literature Review\*\* offers context by discussing the evolution of scoreboard systems, reviewing existing technologies, and identifying gaps in current solutions, particularly in the Nepalese cricket context.

- \*\*Unit 3: System Requirement and Analysis\*\* presents the hardware and software requirements necessary for the system, alongside a feasibility study covering technical, economic, legal, and operational aspects. This unit also includes process and data modeling, providing visual representations of the system's design.

- \*\*Unit 4: Time, Cost, and Task Division\*\* outlines the project timeline, cost estimation, and the division of tasks among team members, ensuring clarity in project management.

- \*\*Unit 5: Implementation and Testing\*\* describes the implementation strategy and testing methodologies, detailing unit and system testing to ensure the reliability of the application.

- \*\*Unit 6: Code and Snapshots\*\* showcases the actual code and user interface snapshots, illustrating the system's functionality and design.

- \*\*Unit 7: Conclusion and Bibliography\*\* summarizes the project's achievements, offers recommendations for future work, and lists references that informed the development of the Cricket Scoreboard system.

This organized approach ensures a logical flow of information, facilitating an understanding of the system's development and its significance in cricket scoring.

**UNIT 2**

**BACKGROUND STUDY AND LITERATURE REVIEW**

### 2.1 Background Study

The background study of the Cricket Scoreboard system explores the historical context and evolution of cricket scoring methods. Traditionally, scoring in cricket was performed manually, requiring scorekeepers to diligently record runs, overs, and wickets using pen and paper. This method, while functional, was susceptible to human error and often resulted in inaccuracies, particularly during fast-paced matches.

With the advancement of technology, various digital scoring systems emerged, offering automated calculations and real-time updates. These systems aimed to improve accuracy and efficiency, yet many still faced challenges, such as complex interfaces and limited accessibility. Additionally, some solutions lacked integration capabilities with other sports applications, restricting comprehensive data analysis and reporting.

In Nepal, cricket has gained popularity, leading to a rise in local tournaments and matches. However, many grassroots organizations continue to rely on outdated scoring methods, which can hinder the development of the sport. This background highlights the necessity for a modern Cricket Scoreboard system that not only enhances accuracy and user experience but also caters specifically to the needs of local cricket environments. The proposed system aims to bridge these gaps by leveraging technology to provide a reliable, user-friendly solution for managing cricket match statistics.

### 2.2 Literature Review

The literature review examines existing systems and technologies related to cricket scoring, highlighting their strengths and weaknesses. Various studies and applications have been developed to address the challenges of manual scoring, with some key observations:

#### 2.2.1 Study of Existing Systems

Several digital scoreboard systems have been implemented globally, ranging from basic mobile applications to comprehensive software platforms used in professional leagues. These systems often feature automated score calculation, real-time updates, and detailed player statistics. However, many existing solutions have limitations, such as steep learning curves, insufficient user interfaces, and lack of support for local or grassroots cricket tournaments.

Additionally, while some systems provide data analytics and reporting features, they often do not integrate well with other sports applications, hindering a comprehensive understanding of player performance over time. Many systems also require consistent internet connectivity, which may not be available in all cricket venues, especially in rural areas.

#### 2.2.2 Gaps Identified in Existing Systems in Nepal

In Nepal, the popularity of cricket has surged, yet the infrastructure for scoring and statistics remains underdeveloped. Most local matches still rely on manual scoring, which is inefficient and prone to errors. Existing digital solutions are either too advanced for grassroots use or do not cater specifically to the unique needs of the local cricketing community. This gap creates an opportunity for a tailored Cricket Scoreboard system that provides essential features while being user-friendly and accessible.

By identifying these gaps, the literature review underscores the need for a new solution that balances functionality with usability, aiming to improve the overall experience of scoring and following cricket matches in Nepal and similar contexts.

**UNIT 3:**

**SYSTEM REQUIREMENT AND ANALYSIS**

**Hardware Configuration:**

The hardware requirements for this program are minimal as it is a simple console-based application written in C. Here's what you would typically need:

1. **CPU:** Any modern processor (Intel, AMD, ARM) would suffice. Even a low-end processor like Intel Core i3 or equivalent would be more than enough.
2. **RAM:** The program requires very little memory, likely under 1 MB of RAM for execution, so even a system with 256 MB of RAM would handle it easily.
3. **Storage:** Since it's a small C program, it needs just a few KBs of disk space to store the source code and compiled binary. Any basic storage (even as low as a few MB) would be more than enough.
4. **Operating System:** The program can run on most operating systems, including:
   * Windows
   * Linux
   * macOS

It will require a C compiler to run, which is available on all of these platforms.

**Software Configuration:**

1. **C Compiler:** To compile and run this C program, you need a C compiler. Some popular options include:
   * **GCC (GNU Compiler Collection):** Available on Linux, macOS, and can be installed on Windows via MinGW or Cygwin.
   * **Clang:** Another open-source compiler for Linux and macOS.
   * **Microsoft Visual C++ (MSVC):** Available on Windows, this is part of the Visual Studio IDE.
   * **Dev-C++:** An integrated development environment (IDE) with a built-in compiler for Windows.
2. **IDE or Text Editor (optional):** While not strictly necessary, using an IDE or a good text editor can make writing and debugging your code easier. Some popular ones include:
   * Visual Studio Code
   * Sublime Text
   * Code: Blocks
   * Dev-C++
   * Eclipse CDT (for C/C++ development)
3. **Operating System:** As mentioned earlier, the program is platform-agnostic. You can run it on any OS with a C compiler, such as:
   * Windows (using MinGW, Cygwin, or Visual Studio)
   * Linux (most distros come with GCC pre-installed)
   * macOS (install Xcode or Command Line Tools for GCC/Clang)
4. **Terminal or Command Line**: After compiling, you'll run the program from a terminal (Linux/macOS) or Command Prompt/PowerShell (Windows).

### 3.3 System Analysis and Feasibility Study

In this section, we conduct a thorough analysis of the Cricket Scoreboard system, evaluating its feasibility across several dimensions: technical, economic, legal, operational, and schedule feasibility.

#### 3.3.1 Technical Feasibility

The technical feasibility assesses whether the required technology can support the development and implementation of the Cricket Scoreboard system. The project will be developed using the C programming language, which is suitable for creating efficient and performant applications. The system will utilize file handling for data storage, ensuring that player statistics and match details can be saved and retrieved effectively. Existing hardware (e.g., personal computers or laptops) will suffice, making it accessible for users in local cricket settings.

#### 3.3.2 Economic Feasibility

Economic feasibility examines the cost-effectiveness of the project. The primary expenses will include development time and potential software tools, which are minimal since the project leverages open-source technologies. Given the expected benefits, such as improved accuracy and enhanced user experience, the project is deemed economically viable. Cost savings from reduced scoring errors and increased efficiency further justify the investment.

#### 3.3.3 Legal Feasibility

Legal feasibility involves ensuring compliance with relevant laws and regulations, including copyright and data protection laws. The Cricket Scoreboard system will not infringe on any proprietary technologies, as it will utilize original coding and design. User data will be handled responsibly, focusing on privacy and compliance with applicable regulations, which is crucial for maintaining user trust.

#### 3.3.4 Operational Feasibility

Operational feasibility assesses how well the system can be integrated into current cricketing practices. The user-friendly interface aims to accommodate scorekeepers of varying skill levels, ensuring a smooth transition from manual to digital scoring. Training sessions and support materials will be provided to facilitate adoption, making the system operationally feasible in various cricket environments.

#### 3.3.5 Schedule Feasibility

Schedule feasibility evaluates whether the project can be completed within a reasonable timeframe. The development process is structured into phases—requirement gathering, design, implementation, testing, and deployment—each with clear milestones. With a dedicated team and a well-defined timeline, the project is expected to be completed efficiently, allowing for timely deployment ahead of local cricket tournaments.

This comprehensive analysis confirms that the Cricket Scoreboard system is technically, economically, legally, operationally, and schedule feasible, paving the way for its successful development and implementation.

### 3.4 Analysis

This section outlines the analysis phase of the Cricket Scoreboard system, focusing on process modeling and data modeling to ensure a structured approach to development.

#### 3.4.1 Process Modeling through DFD

Process modeling utilizes Data Flow Diagrams (DFDs) to represent the flow of information within the Cricket Scoreboard system. The DFDs illustrate how data moves between processes, data stores, and external entities.

1. \*\*Level 0 DFD\*\*: This high-level diagram provides an overview of the system, showcasing the main functions such as adding player details, viewing statistics, and generating match summaries. It identifies the key inputs (user data) and outputs (score reports) of the system.

2. \*\*Level 1 DFD\*\*: This more detailed diagram breaks down each main function into sub-processes. For instance, the "Add Player Details" process is expanded to show the steps for inputting batsman and bowler information, including data validation and storage. This level of detail helps clarify the interactions between various components of the system.

#### 3.4.2 Data Modeling through ER Diagram

Data modeling employs Entity-Relationship (ER) diagrams to define the data structures and relationships within the system. The ER diagram identifies key entities such as Batsman, Bowler, and Match, outlining their attributes and relationships:

1. \*\*Entities\*\*:

- \*\*Batsman\*\*: Attributes include name, runs, balls faced, fours, sixes, and strike rate.

- \*\*Bowler\*\*: Attributes include name, runs given, wickets taken, overs bowled, and economy rate.

- \*\*Match\*\*: Attributes may include date, teams involved, and match results.

2. \*\*Relationships\*\*:

- A Batsman can participate in multiple Matches, while a Match can involve multiple Batsmen.

- A Bowler can also participate in multiple Matches, and similarly, each Match can have multiple Bowlers.

By establishing these entities and their relationships, the ER diagram provides a clear framework for database design, ensuring that data is organized efficiently for retrieval and reporting.

Together, the process modeling and data modeling create a solid foundation for the development of the Cricket Scoreboard system, ensuring clarity in both functionality and data management.

***UNIT 4 : TIME, COST AND TASK DIVISION***



**4.1 .: Time:**

Time, cost and Task Division plays a vital role in the software development. So, the above factors should be properly considered while developing the software. The estimated time duration of this project was around 7 days.

**4.2.: Cost Estimation:**

The cost estimation of a project like the one you're working on can depend on a number of factors, including the size and complexity of the project, the development platform, the development team, and other resources such as software tools and testing. Here's a general breakdown to estimate the cost:

|  |  |  |
| --- | --- | --- |
| **S.N** | **Activities** | **Price(NRs)** |
| 1 | Data collection | 400 |
| 2 | Internet Usage | 1200 |
| 3 | Transportation | 300 |
| 4 | Interface design | 250 |
| 5 | Designing | 200 |
| 6 | Testing | 600 |
| 7 | Miscellaneous Expenses | 250 |
| Total | | 3200 |

*Table no. 1.1: Cost Estimation.*

**4.3: Task Division:**

This project was prepared in a group. The name of the members and their task divisions are mentioned as below;

|  |  |  |
| --- | --- | --- |
| **S.N** | **Name of the Students** | **Tasks Performed** |
| 1. | Ankit Bashyal | Typing Document and Coding |
| 2. | Nirajan Pandey | Presentation |
| 3. | Prince Kharel | Presentation |
| 4. | Ayush Khanal | Rough Sketching and Typing Document |

*Table no. 1.2: Task Division.*

### **13**

**UNIT 5:**

**IMPLEMENTATION AND TESTING**

### 5.1 Implementation Overview

The implementation phase of the Cricket Scoreboard system involves translating the design specifications into a functional software application. This phase encompasses several key activities that ensure the successful deployment of the system:

1. \*\*Environment Setup\*\*:

- The development environment is established, including necessary software tools such as compilers, text editors, and libraries for C programming. This setup ensures that all team members have access to a consistent working environment.

2. \*\*Code Development\*\*:

- The core functionalities of the system are coded according to the design documents. This includes implementing functions for:

- \*\*Data Entry\*\*: Allowing users to input details for batsmen and bowlers, including runs, balls faced, and wickets taken.

- \*\*Data Storage\*\*: Utilizing file handling to save player statistics and match information in a structured format, enabling easy retrieval and updates.

- \*\*Calculations\*\*: Implementing algorithms to calculate key metrics such as runs scored, strike rates, and economies based on user input.

3. \*\*User Interface Design\*\*:

- The user interface is developed to ensure a seamless experience for users. Emphasis is placed on clarity and accessibility, allowing users to navigate through the system easily and perform tasks with minimal effort.

4. \*\*Integration of Components\*\*:

- Individual components of the system, such as data entry, viewing statistics, and generating reports, are integrated to work cohesively. This ensures that data flows smoothly between different parts of the application.

5. \*\*Preliminary Testing\*\*:

- Initial testing is conducted during the implementation phase to identify and rectify any immediate issues in the code. This includes verifying that each function operates correctly and that data is handled accurately.

6. \*\*Documentation\*\*:

- Throughout the implementation, documentation is maintained to capture the code structure, design choices, and any challenges encountered. This documentation serves as a valuable resource for future development and maintenance.

The implementation phase is crucial as it transforms theoretical designs into a working system. By focusing on systematic coding practices, user-friendly design, and thorough documentation, the Cricket Scoreboard system is positioned for successful deployment and user adoption.

### 5.2 Testing Overview

The testing phase is crucial for ensuring the Cricket Scoreboard system operates as intended, providing users with accurate and reliable functionality. This phase encompasses multiple testing methodologies to systematically identify and resolve any issues prior to deployment.

1. \*\*Testing Objectives\*\*:

- The main goal is to validate that all components work correctly and meet the specified requirements. This involves checking for accurate calculations, data integrity, and overall performance.

2. \*\*Types of Testing\*\*:

- \*\*Unit Testing\*\*: Each function is tested individually to confirm that it performs its intended task accurately. For example, functions for saving player details and calculating statistics are evaluated with various inputs to ensure they function correctly.

- \*\*Integration Testing\*\*: After unit testing, the integration of components is tested to ensure that they work together seamlessly. This verifies that data flows correctly between different parts of the system.

- \*\*System Testing\*\*: The complete system undergoes testing to assess its performance as a whole. This includes simulating real-world scenarios to confirm that the application can handle multiple user inputs effectively.

- \*\*User Acceptance Testing (UAT)\*\*: Feedback from actual users, such as scorekeepers and coaches, is collected to ensure the system meets their needs and expectations. This helps identify any usability issues and provides insights for final refinements.

3. \*\*Documentation of Testing\*\*:

- Throughout the testing process, comprehensive documentation is maintained, including test cases, results, and bug reports. This record provides a clear reference for future maintenance and enhancements.

### 5.2.1 Unit Testing

Unit testing focuses on verifying the functionality of individual components or functions within the Cricket Scoreboard system. The key aspects include:

1. \*\*Purpose\*\*:

- To ensure that each function operates correctly in isolation, validating its logic and output for a variety of input scenarios.

2. \*\*Process\*\*:

- Each function, such as those for inputting batsman and bowler details or calculating statistics, is tested with both valid and invalid inputs. This helps confirm that the functions can handle edge cases and produce the expected results.

3. \*\*Tools\*\*:

- While manual testing can be performed, automated testing frameworks can also be employed to facilitate the unit testing process, allowing for more efficient and repeatable tests.

4. \*\*Outcomes\*\*:

- Any defects discovered during unit testing are documented and addressed promptly, ensuring that the individual components are robust before they are integrated into the larger system.

### 5.2.2 System Testing

System testing evaluates the Cricket Scoreboard system as a complete entity, ensuring all components work together harmoniously. Key elements include:

1. \*\*Objective\*\*:

- To assess the overall functionality, performance, and reliability of the system in a real-world context, confirming it meets user requirements.

2. \*\*Process\*\*:

- Comprehensive test cases are developed based on user scenarios, including adding player details, viewing statistics, and generating match summaries. The application is subjected to various load conditions to evaluate its performance under stress.

3. \*\*Types of Tests\*\*:

- Functional testing verifies that the system performs all required functions correctly.

- Non-functional testing evaluates aspects like performance, usability, and security.

4. \*\*Reporting\*\*:

- Any issues identified during system testing are documented in detail, with priorities assigned for resolution. This ensures that critical bugs are addressed before deployment.

By conducting thorough unit and system testing, the Cricket Scoreboard project ensures a high-quality, reliable application that meets the needs of users in the cricket community.

### 7.1 Conclusion

The Cricket Scoreboard system serves as a comprehensive tool for managing and tracking player statistics during cricket matches. Through its user-friendly interface, the application allows users to easily add, view, and summarize data related to both batsmen and bowlers. The key features, such as input validation and detailed statistics calculation, ensure that users can maintain accurate and relevant information for each player.

In conclusion, the Cricket Scoreboard system has transformed the task of tracking player statistics into an engaging and efficient experience, akin to hitting a six on a flat pitch. By integrating C programming, file handling, and user interaction, we've created a tool that assists coaches and players in evaluating performance, making it easier to determine when to practice batting or deliver a motivational pep talk. This project stands as a digital companion, loyally documenting every run and wicket.

Amid the coding challenges and moments of humor, we’ve embraced the emotional essence of cricket—a sport that unites friends, families, and entire nations. Each statistic tells a story, reflecting the excitement and memories woven into every match. Looking ahead, the potential for this system is boundless, and we are eager to enhance it further, ensuring it evolves alongside the game we all love. Here’s to cricket, coding, and the countless runs yet to be scored!

### 7.2 Future Recommendations

To enhance the Cricket Scoreboard system and increase its functionality, several future recommendations can be considered:

1. \*\*User Authentication\*\*:

- Implement user login and authentication features to restrict access to sensitive data and allow multiple users to maintain separate records. This would be particularly useful for teams and coaches managing player statistics.

2. \*\*Graphical User Interface (GUI)\*\*:

- Develop a graphical user interface to improve user experience. A GUI can make navigation more intuitive and visually appealing, allowing users to interact with the system more easily.

3. \*\*Real-Time Updates\*\*:

- Introduce functionality for real-time updates during matches, enabling live tracking of player performances and instant feedback. This could be implemented through a mobile application or web interface.

4. \*\*Advanced Statistical Analysis\*\*:

- Incorporate advanced statistical analysis tools to provide deeper insights into player performances. This could include metrics like batting averages, strike rates over time, and comparison graphs among players.

5. \*\*Data Export Options\*\*:

- Add features to export player statistics and match summaries in various formats (e.g., CSV, PDF). This would facilitate easy sharing of data with teams, coaches, and players for further analysis or reporting.

6. \*\*Integration with Other Systems\*\*:

- Consider integrating the scoreboard system with existing sports management platforms or databases, allowing seamless data flow and synchronization across systems used for coaching, training, and performance analysis.

7. \*\*Multilingual Support\*\*:

- Implement multilingual support to cater to a wider audience. This would make the application accessible to non-English speaking users, broadening its usability.

8. \*\*Feedback Mechanism\*\*:

- Establish a feedback mechanism within the application to gather user suggestions and report issues. This would enable continuous improvement based on user experiences.

By pursuing these recommendations, the Cricket Scoreboard system can evolve into a more robust and versatile tool, ultimately enhancing its value to users and contributing to more effective cricket match management.

### 7.3 Bibliography

The bibliography section provides a list of references and resources that were consulted and utilized during the development of the Cricket Scoreboard system project. This includes books, articles, websites, and other materials that contributed to the understanding of cricket statistics, programming techniques, and software development principles. Below is a sample bibliography that may be relevant:

1. \*\*Books\*\*:

<https://www.researchgate.net/publication/371051540_AN_INSIGHTS_ON_CRICKET_DATA_ANALYTICS/download>

https://progforperf.github.io/Expert\_C\_Programming.pdf

2. \*\*Academic Papers\*\*:

<https://www.researchgate.net/publication/330578956_The_Role_of_Data_Analytics_in_Modern_Day_Sports>

https://www.researchgate.net/publication/345937877\_Sport\_Analytics\_A\_Review

3. \*\*Websites\*\*:

- ESPN Cricinfo. (n.d.). “Statistics.” Retrieved from [www.espncricinfo.com](http://www.espncricinfo.com)

- Stack Overflow. (n.d.). “C Programming Questions and Answers.” Retrieved from [www.stackoverflow.com](http://www.stackoverflow.com)

4. \*\*Documentation and Tutorials\*\*:

- “C Programming Language.” (n.d.). Retrieved from [www.cprogramming.com](http://www.cprogramming.com)

- “File Handling in C.” (n.d.). Tutorialspoint. Retrieved from [www.tutorialspoint.com](http://www.tutorialspoint.com)

5. \*\*Software Development Resources\*\*:

- Pressman, R. S. (2014). \*Software Engineering: A Practitioner's Approach\*. New York: McGraw-Hill Education.

This bibliography serves as a foundational resource for understanding the methodologies, techniques, and frameworks employed in the development of the Cricket Scoreboard system. It acknowledges the contributions of previous works and provides a pathway for further exploration of the topics covered in the project.