

A Project Report

on

Match Master

Submitted in partial fulfilment of the requirements of

Project-VI (BIT356CO)

of

Bachelor of Information Technology

Submitted To



Purbanchal University

Biratnagar, Nepal

Submitted By

Dhiraj Sapkota (360272)

Elisha Rai (360273)

Sagar Upadhyaya (360287)

KANTIPUR CITY COLLEGE

Putalisadak, Kathmandu

September, 2024

A Project Report

On

Match-Master

Submitted in partial fulfilment of the requirements of

Project-VI (BIT356CO)

of

Bachelor of Information Technology

Submitted To

Purbanchal University

Biratnagar, Nepal

Submitted By

Dhiraj Sapkota (360272)

Elisha Rai (360273)

Sagar Upadhyaya (360287)

Project Supervisor

Mr. Saroj Pandey

KANTIPUR CITY COLLEGE

Putalisadak, Kathmandu

September, 2024

Topic Approval Sheet

It is here by informed that the topic selected by Dhiraj Sapkota, Elisha Rai and Sagar Upadhyaya of BIT VI semester project has been found suitable and as per the credit assigned by Purbanchal University (PU), Biratnagar, Nepal. The Project Committee has approved the following topic and supervisor for the mentioned students. This project has been completed for the prescribed period and the project embodied the result of their investigation conducted while they worked as full-time students of this institution.

Topic Approved: Match Master

Mr. Ashim KC

Program Coordinator, Department of IT

Kantipur City College

Mr. Rubim Shrestha

Project Coordinator

Kantipur City College

Certificate From the Supervisor

This is to certify that the project entitled Match Master submitted by Dhiraj Sapkota, Elisha Rai and Sagar Upadhyaya to the Department of Information Technology, School of Science and Technology at Kantipur City College, Putalisadak, Kathmandu towards the requirement for BIT VI project is an original work carried out by them under my supervision and guidance.

Mr. Saroj Pandey

HoD, Department of Information Technology

Kantipur City College

(Project Supervisor)

Acknowledgements

We are very grateful to submit this project report. This report wouldn't have been possible without the immense contribution of a few people. Firstly, we would like to express our deepest gratitude to Purbanchal University for providing us with this opportunity to present this project and its report.

We would like to express our gratitude and deep respect to our respected project supervisor Mr. Saroj Pandey who helped us solve any problems that arose during the preparation of this project.

Lastly, we want to thank Kantipur City College for providing us with this opportunity by approving our project. We are thankful to our dear friends for constant support and encouragement, and we would like to thank our seniors as well who helped us out while we were stuck at a problem during the development of our project. We hope that this project will fulfill the course requirement.

With regards

Dhiraj Sapkota

Elisha Rai

Sagar Upadhyaya

Abstract

Match Master is an innovative project that focuses on predictive analytics within the realm of cricket, specifically using the extensive historical dataset from the Indian Premier League (IPL). The primary objective of the project is to predict match outcomes and inning scores, providing valuable insights into the dynamics of the game. The process begins with the meticulous cleaning and preprocessing of the dataset to remove any inconsistencies or errors, ensuring that the data used for analysis is reliable and accurate.

The core of Match Master lies in its use of advanced machine learning models tailored to forecast the winner of a match and the potential score a team might achieve in an inning. By analyzing data from previous overs, the models are able to capture patterns and trends that are indicative of future performance. This predictive capability is grounded in a thorough understanding of the game and is designed to offer precise and actionable insights for teams, analysts, and fans alike.

Ultimately, Match Master aims to elevate the strategic planning and decision-making process within cricket. By transforming raw data into accurate and insightful predictions, the project not only enhances the understanding of the game but also offers a powerful tool for those looking to gain a competitive edge in the IPL. Whether for professional teams or enthusiastic followers of the sport, Match Master represents a significant advancement in the application of data science to cricket.

Table of Content

Chapter 1: Introduction	1
1.1 Overview	1
1.2 Problem Statement	1
1.3 Objectives	2
1.4 Features	2
1.5 Significance	2
1.6 Scope and Limitation	3
1.6.1 Scope.....	3
1.6.2 Limitation	4
Chapter 2: Literature Review	5
Chapter 3: Methodology.....	7
3.1 Software Development Life Cycle.....	7
3.2 Technologies and Tools	9
3.2.1 Specified Programming Language.....	9
3.2.2 Specified Dataset	9
3.2.3 Specified Software	9
3.3 Assignments of Rolls and Responsibilities	9
Chapter 4: System Analysis.....	10
4.1 Requirement Analysis	10
4.1.1 Requirement Gathering	10
4.1.2 Functional Requirement.....	11
4.1.3 Non-Functional Requirement.....	12
4.2 Feasibility study.....	13
4.2.1 Technical Feasibility	13
4.2.2 Economic Feasibility	13
4.2.3 Operational Feasibility	13
4.2.4 Schedule Feasibility	14
Chapter 5: System Design	15

5.1 System Architecture.....	15
5.2 Data Flow diagram.....	15
5.2.1 Context Level (level 0) DFD:	15
5.2.2 Level 1 DFD:	16
5.2.3 Use Case Diagram:	17
5.3 Activity Diagram	18
5.4 Database Design.....	19
5.4.1 ER Diagram	19
5.4.2 Data Dictionary	20
Chapter 6: System Development and Implementation.....	22
6.1 Programming Platform.....	22
6.2 Operating Environment.....	22
6.3 Functional Implementation	23
Chapter 7: Testing and Debugging	24
7.1 Tools used in testing.	24
7.2 Test Cases.....	25
Chapter 8: Conclusion	27
Chapter 9: Future Enhancement.....	28
REFERENCES.....	29
APPENDIX.....	30

List of figures

Figure No.	Name	Page No.
3.1	Waterfall Model	7
4.2.4.1	Gantt Chart	14
5.1	System Architecture	15
5.2.1	Level 0 DFD	15
5.2.2	Level 1 DFD	16
5.2.3	Use Case Diagram	17
5.3	Activity Diagram	18
5.4.1	E-R Diagram	19

List of tables

Table No.	Name	Page No.
3.3	Assignment of Rolls and Responsibilities	9
4.1.2	Functional Requirements	11
4.1.3	Non-Functional Requirements	12
5.4.2.1	Data for Score	20
5.4.2.2	Data for Win	21
7.1	Tools used in Testing	24
7.2.1	Test Case	25 & 26

Chapter 1: Introduction

Match Master is a project that uses data analysis to predict the outcomes of cricket matches. By looking at a large amount of historical data, it aims to forecast which team will win and how many runs a team might score in an inning, based on previous overs. The project starts by cleaning the data to remove any errors, ensuring that the information used is accurate. Then, advanced models are applied to make predictions. Match Master is designed to help teams, analysts, and fans understand the game better and make more informed decisions.

1.1 Overview

Match Master is designed to predict cricket match outcomes and inning scores using a comprehensive historical dataset. The project starts with a detailed data-cleaning process to eliminate inaccuracies, ensuring the data's integrity. Advanced machine learning models are then used to analyze patterns from previous overs, enabling accurate forecasts of a team's potential score and the match winner. This approach offers valuable insights for teams, analysts, and fans, helping them make informed decisions and better understand the game's dynamics. Match Master transforms raw data into actionable insights, making it a powerful tool in cricket analytics.

1.2 Problem Statement

1. Difficulty in processing and analyzing large volumes of historical cricket match data.
2. Inadequate methods for handling real-time data input and output for match predictions.
3. Limited accuracy of IPL match Predictions due to missing venue
4. Lack of Interpretability of prediction models

1.3 Objectives

To predict:

- Cricket Match Score
- Winning probability of the team

1.4 Features

1. Real-time Predictions
 - Match Momentum Tracking
2. Historical Data Analysis
 - Match Outcome
 - Player Statistics
 - Team Performance
 - Venue & more
3. Interactive and data rendering

1.5 Significance

1. **Enhanced Decision-Making:** Match Master provides teams, coaches, and analysts with data-driven predictions, enabling more strategic decisions during matches.
2. **Improved Fan Engagement:** By offering insights into match outcomes and scores, the project enhances fan experience, making the game more interactive and engaging for enthusiasts.

3. **Data-Driven Strategy:** It bridges the gap between traditional cricket strategies and modern data science, introducing innovative ways to interpret and act upon historical game data.
4. **Competitive Edge:** Teams can use the predictions to gain a tactical advantage, potentially altering game plans to improve performance based on predicted outcomes.
5. **Application of Machine Learning in Sports:** The project showcases the potential of applying machine learning to sports analytics, providing a model for similar innovations across other sports.

1.6 Scope and Limitation

1.6.1 Scope

Match Master is designed to predict match outcomes and inning scores in IPL cricket using advanced machine learning models. The project encompasses the following key areas:

1. **Data Cleaning & Preprocessing:** Ensuring the dataset is accurate by removing inconsistencies for reliable predictions.
2. **Predictive Analytics:** Developing machine learning models that analyze historical IPL data to predict the winner and score of a match.
3. **Insight Generation:** Providing actionable insights for teams, analysts, and fans by identifying trends in match performance.
4. **Strategic Decision-Making:** Offering a data-driven tool to enhance the strategic planning process for cricket teams.

This scope highlights the project's ability to turn raw data into valuable insights, contributing to more informed decisions in cricket.

1.6.2 Limitation:

1. **Data Dependency:** The accuracy of predictions heavily relies on the quality and completeness of the IPL dataset. Missing or incorrect data can lead to less reliable outcomes.
2. **Limited to Historical Patterns:** Machine learning models predict based on past data, so unprecedented situations (e.g., injuries, weather conditions) may not be accounted for effectively.
3. **Model Complexity:** Building and tuning machine learning models can be complex and may require continuous refinement for accuracy, particularly with dynamic game variables.
4. **Generalization to Other Leagues:** Since the model is tailored to IPL data, it may not generalize well to other cricket leagues without significant modifications.

Chapter 2: Literature Review

Predictive analytics has transformed cricket decision-making, with platforms like CricViz and Predict21 offering web-based systems to forecast match outcomes. These models highlight the growing role of data-driven insights in cricket. However, the sport's unpredictability, due to factors like pitch behavior, weather, and momentum shifts, challenges model accuracy. Traditional static models often fall short, prompting the need for adaptive models that update predictions in real-time. While advancements in machine learning have improved prediction capabilities, the inherent uncertainty of cricket requires further innovation in model design and application. To overcome the challenges, we are using large amounts of datasets and following predictive machine learning algorithms.

1. Logistic regression⁴

Logistic regression is defined as a supervised machine learning algorithm that accomplishes binary classification tasks by predicting the probability of an outcome, event, or observation. It is ideal for cricket winning prediction because:

- a) Binary classification: it predicts win/loss a binary outcome
- b) Probability estimates: provide the likelihood of team winning
- c) Efficiency: fast to train and predict, even with small datasets
- d) Baseline models: serves as a benchmark to compare with more complex models

2. Random Forests⁵

Random Forests combines the output of multiple decision trees to reach a single result. Its ease of use and flexibility have fueled its adoption, as it handles both classification and regression problems. We can use for cricket matches win predictions because:

- a) Handle complexity: effectively models complex relationships between variables

- b) High accuracy: Aggregates multiple decision trees for better accuracy
- c) Reduces overfitting: Robust to overfitting by averaging across many trees
- d) Feature importance: identifies key factors influencing match outcomes
- e) Scalable: handles large datasets and many feature efficiently

3. XGBoost regression⁶

XGBoost is a highly efficient and scalable implementation of the gradient boosting algorithm, widely used for its ability to handle large data sets and complex models. It builds models sequentially to correct errors from previous models, with features like regularization, parallel processing, and handling of missing data to enhance performance and prevent overfitting. XGBoost is used for predicting cricket score calculation in this project because:

- a) High accuracy: it effectively models complex relationships
- b) Efficiency: handle large dataset quickly.
- c) Feature importance: identifies key factors impacting scores.
- d) Handles missing data: effectively manages incomplete data

Chapter 3: Methodology

3.1 Software Development Life Cycle

We have used waterfall model to work on this project Match Master is the shortest period or one semester project, it is a small project and requirement are predefined as well as we don't need to move back so we have chosen it as the best model for our project.

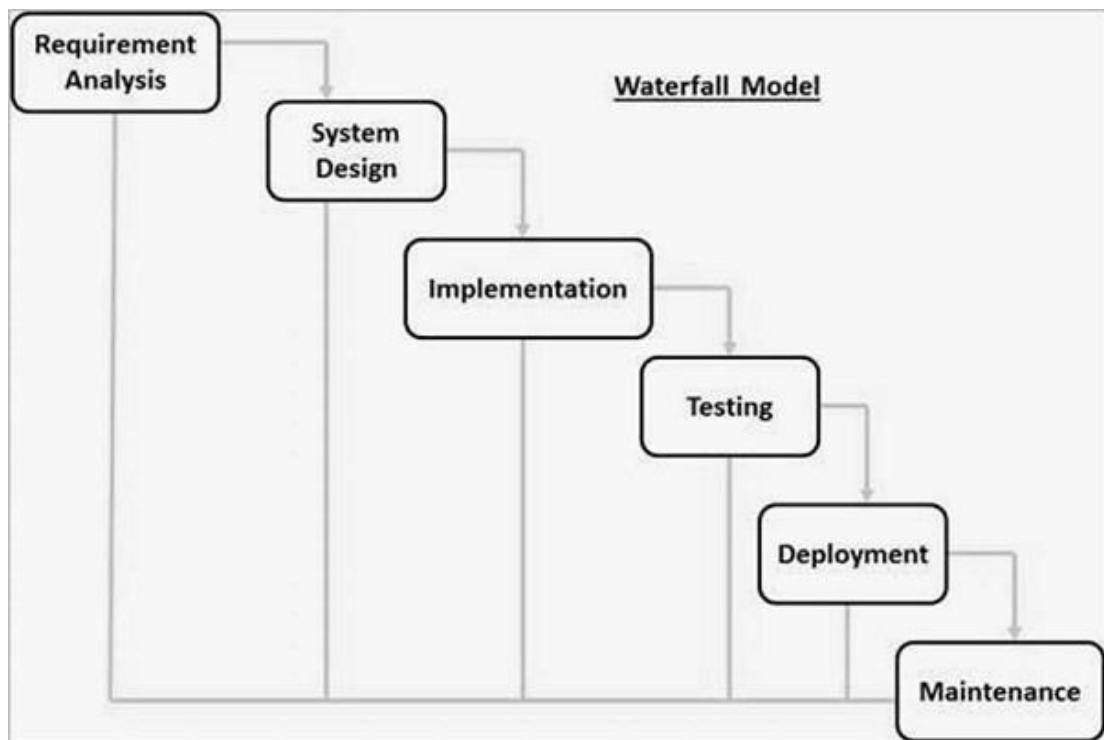


Fig 3.1: Waterfall Model

1. Requirements Gathering and Analysis:

In this initial phase, we all development team gathered and documented all the project requirements. Detailed analysis of the requirements is performed to ensure a clear understanding of the “Match Master” scope and objectives.

2. System Design:

Once the requirements are well-defined, the system design phase begins. In this phase, we designed an overall blueprint of system design. Make an entity-relationship diagram to describe relationships between entity and discussed about attributes of each entity, and design Dataflow Diagram (DFD) to understand flow of data.

3. Implementation (Coding):

In this phase, we started coding based on the design specifications. The code is written and reviewed to ensure it aligns with the design and meets the specified requirements.

4. Testing:

After coding, we are entering the testing phase. We conducted various tests, including unit testing, integration testing, system testing, and user acceptance testing, to identify and rectify defects. Each test is successfully run, after rectifying and debugging errors.

5. Maintenance:

The maintenance phase is the last stage of the waterfall model, where we monitor, update, and fix it after it is released to the users. In this phase we can work as an activity such as bug fixing, performance improvement, security enhancement, and feature addition.

3.2 Technologies and Tools

Match master is a web-based system. In this project we are using various tools and technologies.

3.2.1 Specified Programming Language

Programming language: html 5, CSS 3, JavaScript, python (NumPy, pandas, Scikit Learn, flask)

3.2.2 Specified Dataset

Dataset: CSV file (from Kaggle)

3.2.3 Specified Software

Software: Vs-code, Jupiter notebook

3.3 Assignments of Rolls and Responsibilities

The member assigned with these responsibilities:

Team Members	Task Performed
Dhiraj Sapkota	Requirement Gathering, testing, Coding & Documentation
Elisha Rai	System Analysis & Design, Coding & Documentation
Sagar Upadhyaya	System Analysis & Design, testing, Coding & Debugging

Table 3.3: Assignments of Rolls and Responsibilities

Chapter 4: System Analysis

4.1 Requirement Analysis

In the requirement analysis phase, the collected requirements are examined and organized to define both functional and non-functional aspects of the system. Functional requirements focus on the system's capabilities, such as providing accurate real-time updates, generating detailed predictive models, and integrating interactive visualizations. Non-functional requirements address system performance, scalability, and user interface design to ensure the system operates efficiently and is user-friendly. This phase also includes identifying potential challenges, such as data quality issues and computational limitations, and developing strategies to overcome these challenges. The goal is to ensure that the data-model measures accurate, and actual gaming performance and delivers valuable insights effectively.

4.1.1 Requirement Gathering

The requirement gathering phase for the Match Master project involves systematically identifying the needs and expectations of various stakeholders, including cricket teams, analysts, and fans. This process starts with collecting detailed input through interviews, surveys, and workshops to understand what types of predictions and features are most valuable. Key requirements include real-time predictions of match outcomes, inning scores, and interactive tools for fan engagement. The team also identifies necessary data sources, such as historical match records, player statistics to support accurate and insightful predictions.

4.1.2 Functional Requirement

Requirement Name	Description
Real-Time Prediction	<ul style="list-style-type: none">• The system must provide live updates on predicted match outcomes and inning scores as the match progresses.
Win Probability Model	<ul style="list-style-type: none">• the system should calculate and display the probability of each team winning the match based on current data.
Winning Score Prediction	<ul style="list-style-type: none">• The system must predict potential scores for each inning based on historical performance data and current match conditions.
Interactive Visualizations	<ul style="list-style-type: none">• The system should include dynamic charts and graphs to visualize team performance trends, score projections, and match simulations.

Table 4.1.2: Functional Requirement

4.1.3 Non-Functional Requirement

Requirement Name	Description
Performance	<ul style="list-style-type: none">• Responsive and capable of handling large data.• deliver real-time predictions and updates with minimal latency
Scalability	<ul style="list-style-type: none">• capable of handling large volumes of data and user interactions
Usability	<ul style="list-style-type: none">• Intuitive user interface with clear navigation and user-friendly features
System Interface	<ul style="list-style-type: none">• Describe interfaces for user interaction.
Data Management	<ul style="list-style-type: none">• Defines the database schema and data models• Specify data backup and recovery procedures to ensure data integrity.

Table 4.1.3: Non-Functional Requirement

4.2 Feasibility study

4.2.1 Technical Feasibility

The "Match Master" project is technically feasible given the availability of large historical datasets from the Indian Premier League (IPL). The project will employ machine learning algorithms and data processing techniques that are well-supported by existing technology stacks. Tools like Python, NumPy, and pandas, along with cloud-based solutions provide the necessary computational power and data storage capabilities. The team's proficiency in data science and machine learning ensures that the project can be developed, tested, and deployed effectively. Furthermore, the use of scalable models ensures that as more data becomes available, the system can adapt without significant overhauls.

4.2.2 Economic Feasibility

Economically, "Match Master" presents a viable investment. The primary costs involve data acquisition, computational resources, and personnel for development and maintenance. Given the widespread interest in IPL and sports analytics, the potential return on investment is high. The tool could be monetized through subscription services, partnerships with sports analytics firms, or licensing agreements with teams and broadcasters. Moreover, the cost of cloud services can be optimized by using pay-as-you-go models, ensuring that expenses align with usage and demand.

4.2.3 Operational Feasibility

From an operational standpoint, "Match Master" is feasible as it leverages existing infrastructure and industry-standard practices in machine learning and data management. The project's operations can be integrated into existing sports analytics platforms, providing seamless access for users. Regular updates based on new data will ensure the system remains current, while user-friendly interfaces will make the tool accessible to a broad audience, including non-technical users.

4.2.4 Schedule Feasibility

The project timeline is realistic, with an estimated development period of 6 to 12 months, depending on the complexity of the models and the scale of data preprocessing required. Initial phases will involve data collection and cleaning, followed by model development, testing, and iteration. A phased approach allows for the early release of a minimum viable product (MVP), providing basic prediction capabilities while more advanced features are developed. Continuous integration and deployment (CI/CD) practices will facilitate regular updates and improvements, ensuring the project stays on track and adapts to any unforeseen challenges.

4.2.4.1 Gantt chart

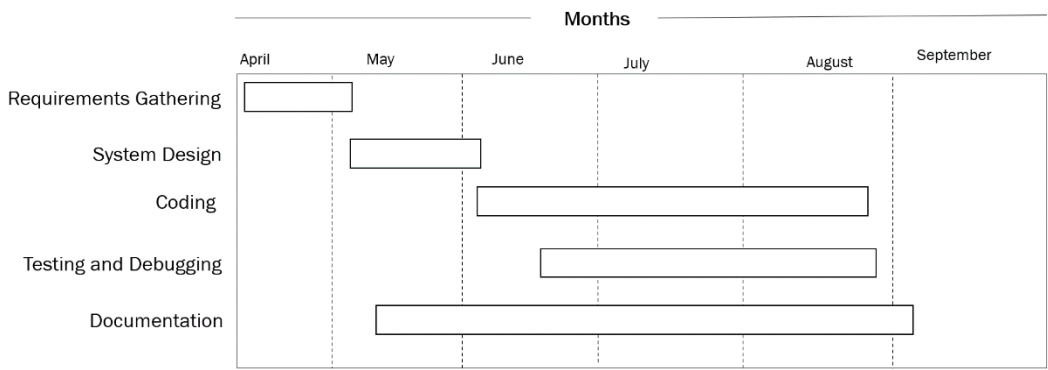


Table 4.2.4.1: Gantt chart

Chapter 5: System Design

5.1 System Architecture

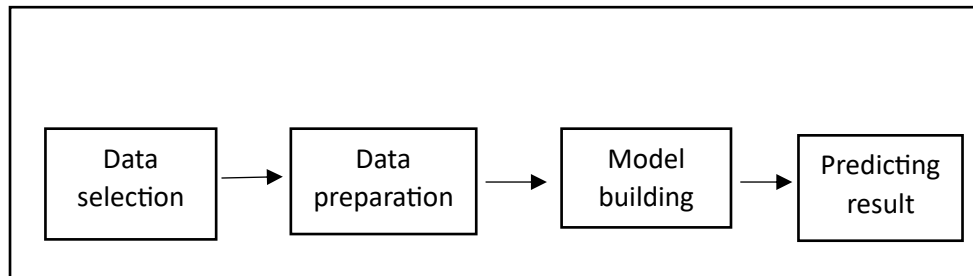


Fig 5.1: Proposed System Architecture

5.2 Data Flow diagram

A data-flow diagram is a way of representing a flow of data through a process or a system.

5.2.1 Context Level (level 0) DFD:

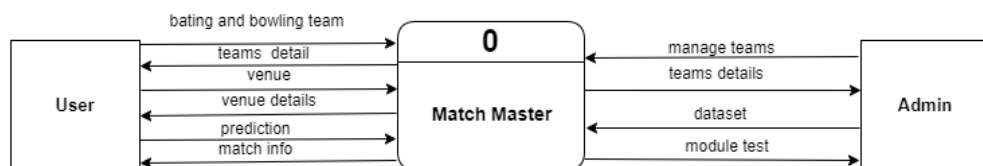


Fig 5.2.1: Level 0 DFD

5.2.2 Level 1 DFD:

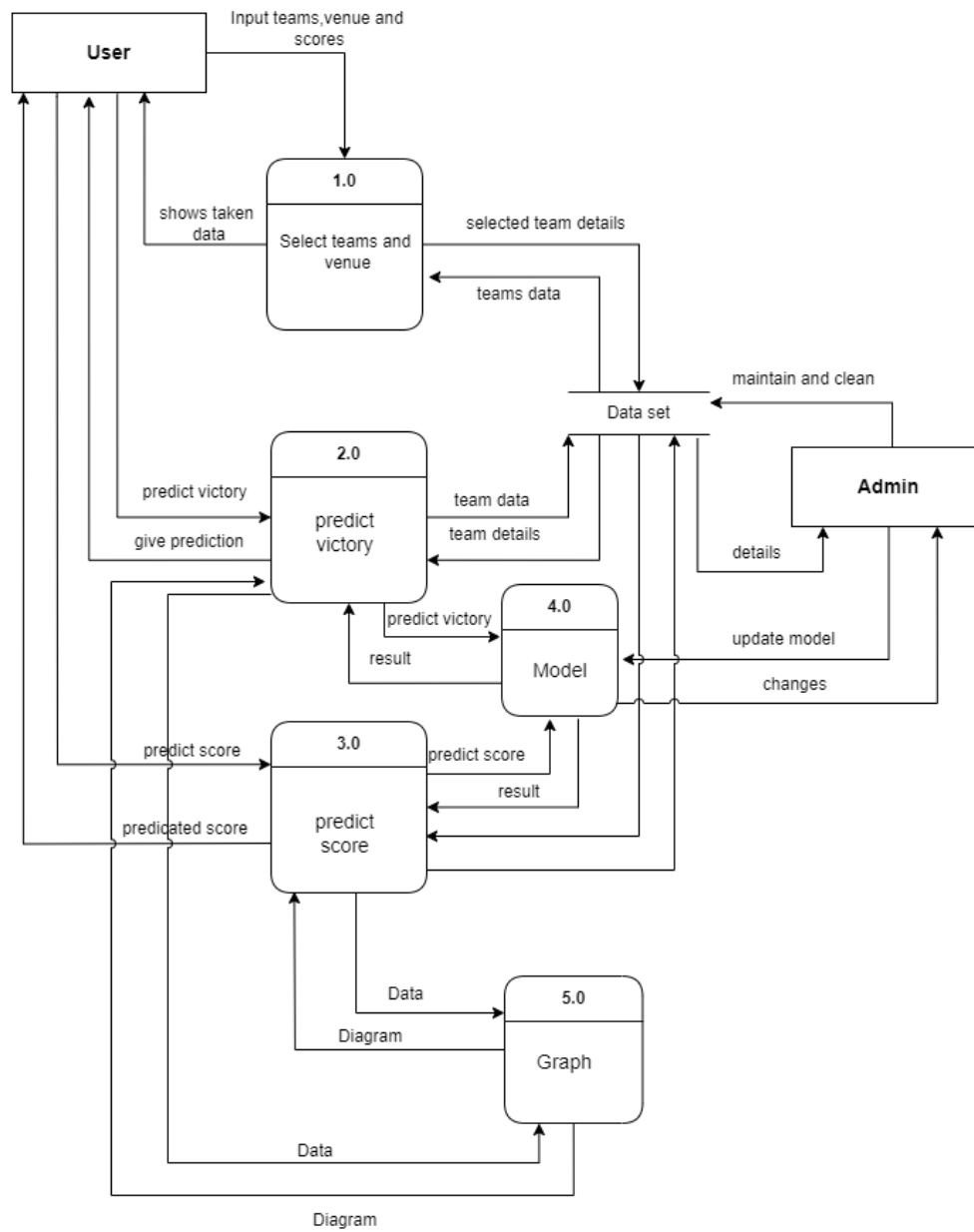


Fig 5.2.2: Level 1 DFD

5.2.3 Use Case Diagram:

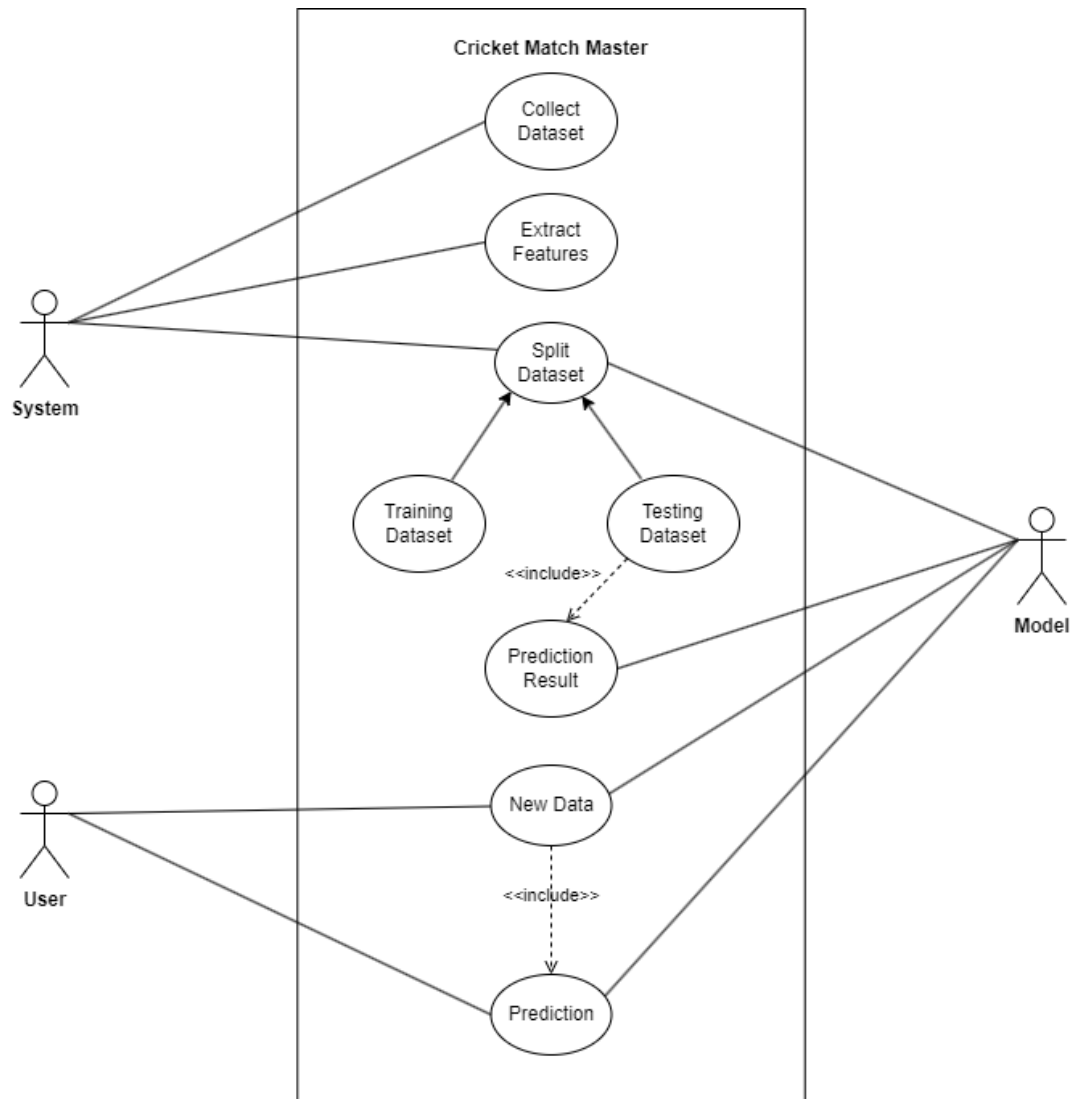


Fig 5.2.3: Use Case Diagram

5.3 Activity Diagram

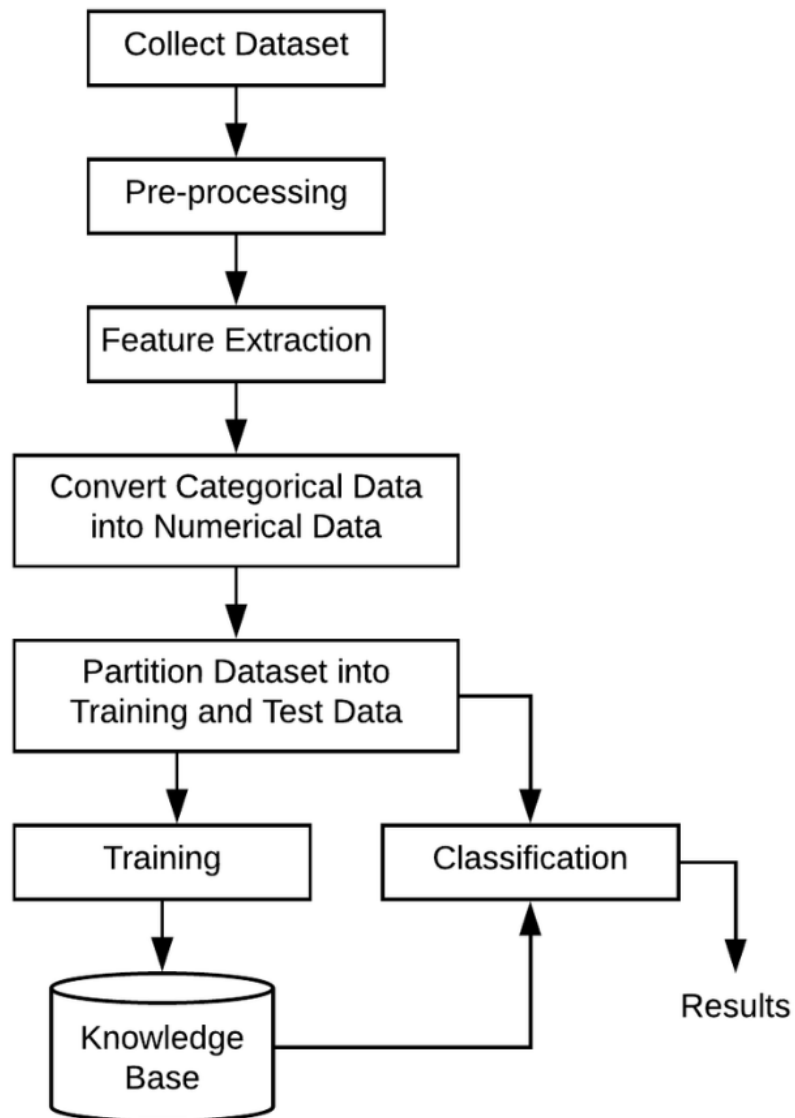


Fig 5.3: Activity Diagram

5.4 Database Design

5.4.1 ER Diagram

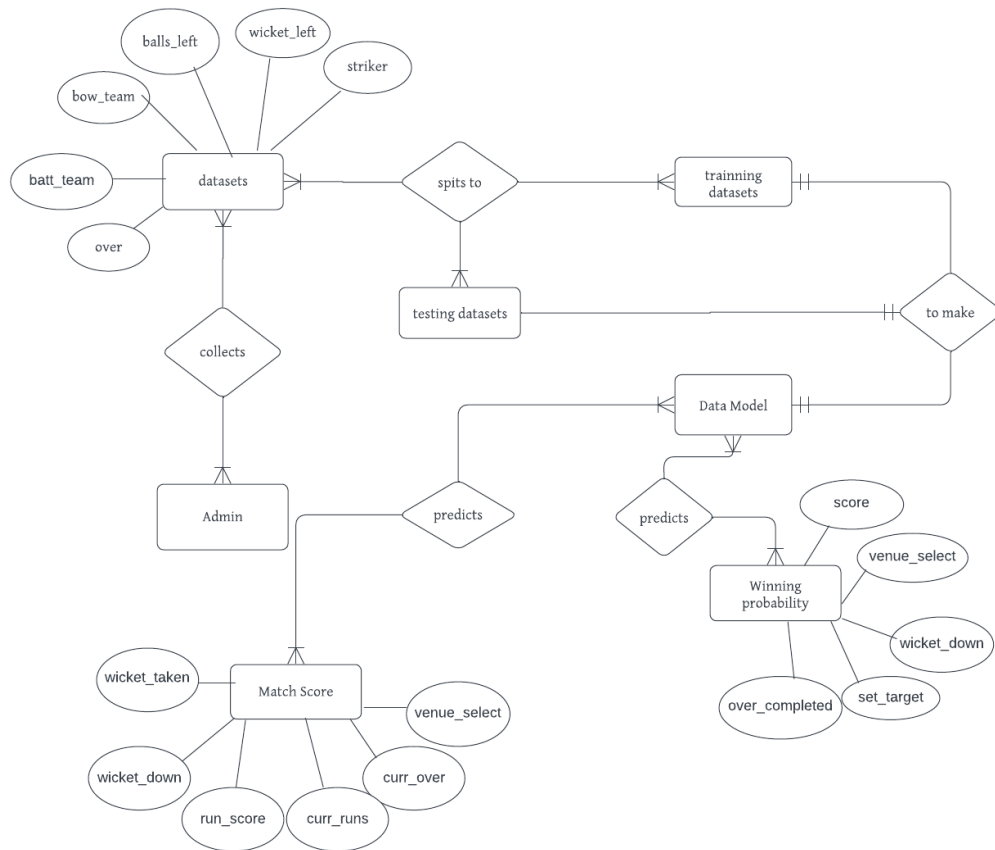


Fig 5.4.1: E-R Diagram

5.4.2 Data Dictionary

5.4.2.1 Data for score

Field Name	Data Type
match_id	int
season	date
venue	str
start_date	date
innings	int
balls	int
batting_team	str
bowling_team	str
cur_score	int
crr	int
balls_left	int
wicket_left	int
last_5_ov	int
city	str
total_score	int

Table 5.4.2.1: Data for score

5.4.2.2 Data for win

Field Name	Data Type
Batting_team	str
Bowling_team	str
city	str
Runs_left	int
Balls_left	int
Wickets_left	int
Total_score	int
crr	int
rrr	int
result	int

Table 5.4.2.2: Data for win

Chapter 6: System Development and Implementation

6.1 Programming Platform

Python is ideal for cricket analytics because it offers a rich ecosystem of libraries like Pandas and NumPy for data manipulation, and scikit-learn for machine learning. Python's ease of integration with databases and web applications allows for seamless automation and real-time analysis. The language's simplicity and strong community support make it accessible and versatile, enabling quick development and scalability for analytics projects.

6.2 Operating Environment

Software Specifications

Computer software specification we have used for development:

- Operating System: Windows 10/11
- Vs-code, Jupiter Notebook
- MS excels

Hardware Specifications

Computer hardware specification we have used for development:

- Processor: Intel Core i5/Ryzen 7
- RAM: 8GB/16GB
- SSD: 512GB

6.3 Functional Implementation

During the functional implementation of Match Master, we utilized HTML, CSS, JavaScript, and python to build the platform's features. HTML and CSS were employed for structuring and styling the user interface to ensure its visually appealing and easy to navigate. JavaScript enhanced user interactions and dynamic content, while Python handled server-side processing and logic, ensuring smooth functionality. Through the integration of these technologies, we successfully developed Match Master, providing a user-friendly platform for predicting win probability and total team score.

Chapter 7: Testing and Debugging

Testing and debugging are essential stages in the development process of Match Master to ensure its reliability and functionality. Testing involves systematically examining each component of the platform to verify that it performs according to its specifications. This process includes unit testing, where individual modules are tested in isolation to ensure they function correctly, and integration testing, where different modules are combined and tested together to ensure they work seamlessly as a whole. Additionally, user acceptance testing is conducted to evaluate the platform from the end-users' perspective, ensuring it meets their needs and expectations. Throughout the testing phase, any defects or issues identified are logged and addressed through debugging. Debugging involves identifying and fixing errors in the code, whether they are logical errors, syntax errors, or other issues preventing the platform from functioning as intended. By conducting thorough testing and debugging, we ensure that Match Master operates reliably and delivers a seamless user experience for cricket fans.

7.1 Tools used in testing.

S. N	Tools	Specification
1	Laptop	Hardware
2	Vs-code	IDE
3	Jupyter Notebook	IDE
4	MS excels	Data store

Table 7.1 Tools used in testing.

7.2 Test Cases

ID	Test case Description	Test Case Data	Excepted Result	Actual Result	Status
T01	Verify Data processing removes duplicates and nulls	IPL dataset containing duplicates and null values	All duplicates and null values should be removed from the dataset	Duplicates removed, but null values still present	Failed
T01-1	Verify null values are removed after preprocessing fix	IPL dataset containing null values	Null values should be successfully removed	Null values removed	Passed
T02	Validate model can predict match outcome based on historical data	Preprocessed dataset from IPL	Model should predict correct match outcome for a set of historical matches	Predicted match outcomes match the actual outcomes	Passed
T03	Validate output for user-friendly visualization	Prediction results output	Results should be displayed in chart format	Predictions displayed in chart format	Passed
T04	Test model performance under high data load	Dataset with 100000+ match records	Model should process large datasets without significant delays	Model took too long to process the data	Failed

T04-1	Optimize model performance under high data load	Live data from an ongoing IPL match	Model should update predictions in real time as new overs are played	Predictions updated in real time	Passed
T05	Check prediction during live match	Live data from an ongoing IPL match	Model should update predictions in real time as new overs are played	Predictions updated in real time	Passed

Table 7.2 Test case

Chapter 8: Conclusion

The "Match Master" project successfully leverages machine learning algorithms to predict cricket match outcomes, specifically focusing on the projected score and the win probability of competing teams. By integrating historical data, player statistics, and various match conditions, the system can generate insightful predictions that can be valuable to fans, analysts, and stakeholders in the cricketing world.

Throughout the development of "Match Master," a comprehensive approach was taken to preprocess the data, select appropriate features, and train robust models capable of making accurate predictions. The results demonstrate that machine learning can indeed be a powerful tool in sports analytics, offering a glimpse into potential outcomes based on empirical data.

In conclusion, "Match Master" not only enhances the experience of cricket enthusiasts by providing predictive insights but also sets a foundation for future advancements in sports prediction models. With further refinement and the inclusion of more diverse datasets, the system could evolve into an even more sophisticated tool, potentially applicable to other sports as well.

Chapter 9: Future Enhancement

1. **Weather & Pitch Analysis:** Incorporate weather conditions and pitch characteristics into the prediction model for improved accuracy.
2. **Mobile App Version:** Develop a mobile application to make predictions accessible on handheld devices for a wider audience.
3. **Explainable AI:** Provide transparent reasoning behind predictions, showing which factors influenced the outcome.
4. **Multilingual Support:** Enable the system to support multiple languages for a more diverse user base.
5. **Player and Team Form Impact:** Factor in current form and recent performance of players and teams to enhance prediction accuracy.

REFERENCES

1. Saji, B. (2024, May 28). *IPL Team Win Prediction Project Using Machine Learning*. Analytics Vidhya.
<https://www.analyticsvidhya.com/blog/2022/05/ipl-team-win-prediction-project-using-machine-learning/>
2. Shukla, P. (2022, October 31). *Building an IPL Score Predictor – End-To-End ML Project*. Analytics Vidhya.
<https://www.analyticsvidhya.com/blog/2021/10/building-an-ipl-score-predictor-end-to-end-ml-project/>
3. Pattanik, S. (2024, March 13). *AI powered cricket analysis*.
<https://medium.com/@pattnaiksatyajit89/ai-powered-cricket-analytics-revolutionizing-ipl-win-predictions-with-ml-bd2d144b59f2>
4. GeeksforGeeks. (2024b, June 20). *Logistic Regression in Machine Learning*. GeeksforGeeks. <https://www.geeksforgeeks.org/understanding-logistic-regression/>
5. *Machine Learning Random Forest Algorithm - Javatpoint*.
[www.javatpoint.com](https://www.javatpoint.com/machine-learning-random-forest-algorithm). <https://www.javatpoint.com/machine-learning-random-forest-algorithm>
6. GeeksforGeeks. (2023, January 10). *XGBoost for Regression*. GeeksforGeeks.
<https://www.geeksforgeeks.org/xgboost-for-regression/>

APPENDIX



Fig: Home Page

Select Batting Team

Kolkata Knight Riders

Select Bowling Team

Delhi Capitals

Select Venue

Ahmedabad

Target

189

Score

112

Current Over

10.0

Wickets Down

3

Predict Winning Probability

Kolkata Knight Riders Winning Probability: 73.0%

Delhi Capitals Winning Probability: 27.0%

View Chart

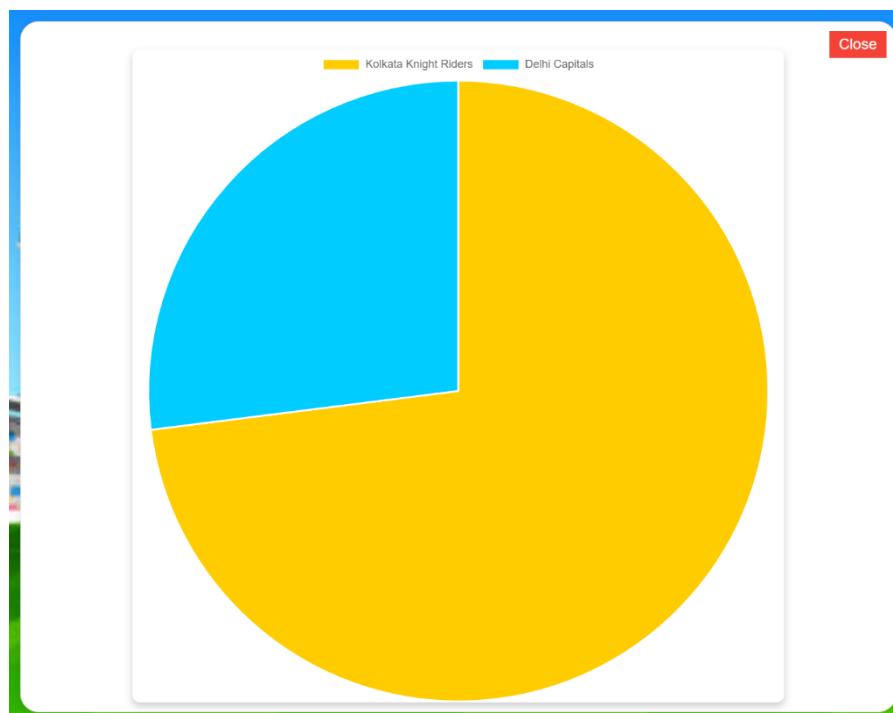


Fig: Win Prediction

Select Batting Team

Chennai Super Kings

Select Bowling Team

Mumbai Indians

Select Venue

Bengaluru

Current Score

85

Current Over

9.0

Wickets Down

2

Last 5 Overs Score

50

Last 5 Overs Wickets

1

Predict Score

Predicted Score: 191

View Score Chart



Fig: Score Prediction