A

Mini Project

On

Predicting Accuracy of Players in the Cricket using Machine Learning

Submitted in partial fulfillment of the requirements for the award of Degree

of

BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE AND ENGINEERING

By

SK. Nazeer [217R1A05Q8]

S. Naresh [217R1A05Q4]

P. Dheeraj [217R1A05P9]

Under the Guidance of

Dr. D.T.V. Dharmajee Rao

Professor & Dean Academics

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CMR TECHNICAL CAMPUS

UGC AUTONOMOUS

(Accredited by NAAC, NBA, Permanently Affiliated to JNTUH, Approved by AICTE, New Delhi)
Recognized Under Section 2(f) & 12(B) of the UGCAct.1956,
Kandlakoya (V), Medchal Road, Hyderabad-501401.

October, 2024

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project entitled "**PREDICTING ACCURACY OF PLAYERS IN THE CRICKET USING MACHINE LEARNING"** being submitted by **SK. NAZEER (217R1A05Q8), S. NARESH (217R1A05Q4)** and **P. DHEERAJ (217R1A05P9)** in partial fulfillment of the requirements for the award of the degree of B.Tech in Computer Science and Engineering to the Jawaharlal Nehru Technological University Hyderabad, is a record of bonafide work carried out by them under our guidance and supervision during the year 2024-25.

The results embodied in this project have not been submitted to any other University or Institute for the award of any degree or diploma.

Dr. D. T. V. Dharmajee Rao
Professor & Dean Academics
INTERNAL GUIDE

Dr. N. Bhaskar
HOD

Dr. A. Raji Reddy
DIRECTOR

EXTERNAL EXAMINER

Submitted for viva voice Examination held on _____

ACKNOWLEDGEMENT

Apart from the efforts of us, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

We take this opportunity to express my profound gratitude and deep regard to our guide **Dr. D. T. V. Dharmajee Rao,** Professor & Dean Academics for his exemplary guidance, monitoring and constant encouragement throughout the project work. The blessing, help and guidance given by him shall carry us a long way in the journey of life on which we are about to embark.

We also take this opportunity to express a deep sense of gratitude to Project Review Committee (PRC) Coordinators : **Dr. J. Narasimha Rao , Dr. K. Maheshwari, Mr. K. Ranjith Reddy, Mrs. K. Shilpa** for their cordial support, valuable information and guidance, which helped us in completing this task through various stages.

We are also thankful to **Dr. N. Bhaskar,** Head of the Department of Computer Science and Engineering for providing encouragement and support for completing this project successfully.

We are obliged to **Dr. A. Raji Reddy,** Director for being cooperative throughout the course of this project. We would like to express our sincere gratitude to Sri. **Ch. Gopal Reddy,** Chairman for providing excellent infrastructure and a nice atmosphere throughout the course of this project.

The guidance and support received from all the members of **CMR Technical Campus** who contributed to the completion of the project. We are grateful for their constant support and help.

Finally, we would like to take this opportunity to thank our family for their constant encouragement, without which this assignment would not be completed. We sincerely acknowledge and thank all those who gave support directly and indirectly in the completion of this project.

SK. NAZEER [217R1A05Q8]
S. NARESH [217R1A05Q4]
P. DHEERAJ [217R1A05P9]

ABSTRACT

Predicting the performance and accuracy of cricket players is a challenging task due to the complex nature of the sport, which involves various skills such as batting, bowling, and fielding. In recent years, machine learning techniques have increasingly been employed to analyze cricket player data and forecast their future performance. This study proposes a machine learning-based approach to predict the accuracy of cricket players. The dataset used in this research comprises historical player statistics, including batting average, bowling average, strike rate, economy rate, and fielding statistics. Various machine learning algorithms, such as linear regression, decision trees, random forests, and support vector machines, are employed to develop predictive models. Feature selection techniques are also applied to identify the most relevant attributes for prediction. The performance of each model is evaluated using metrics such as mean absolute error, mean squared error, and R-squared value. The experimental results demonstrate the effectiveness of the proposed approach in accurately predicting the performance of cricket players. The findings of this study can be valuable for cricket coaches, team selectors, and analysts in making informed decisions regarding player selection, strategy formulation, and performance improvement. Additionally, the methodology presented in this research can be extended to other sports for performance prediction and analysis.

LIST OF FIGURES

LIST OF SCREENSHOTS

TABLE OF CONTENTS

1. INTRODUCTION

1.INTRODUCTION

1.1 PROJECT SCOPE

The proposed system lays the groundwork for future research and development in cricket analytics by allowing for the refinement and optimization of machine learning algorithms for player accuracy prediction, including the exploration of advanced techniques like deep learning and ensemble methods to enhance model robustness. Additionally, it can be extended to incorporate diverse data sources and performance metrics, such as player biometrics, match conditions, and situational analysis, providing a comprehensive understanding of player accuracy and enabling more nuanced predictions tailored to specific game scenarios.

1.2 PROJECT PURPOSE

The purpose of this project is to develop a machine learning-based approach for predicting the performance and accuracy of cricket players by analyzing historical player statistics. By employing various machine learning algorithms and feature selection techniques, the project aims to create predictive models that can assist coaches, team selectors, and analysts in making informed decisions about player selection, strategy formulation, and performance improvement. Ultimately, the methodology can also be adapted for use in other sports to enhance performance prediction and analysis.

1.3 PROJECT FEATURES

This project leverages a comprehensive dataset of historical player statistics, including key metrics like batting and bowling averages, strike rates, and fielding statistics. It employs various machine learning algorithms—such as linear regression, decision trees, random forests, and support vector machines—for effective model comparison. Advanced feature selection enhances accuracy by identifying relevant attributes, while performance is evaluated using metrics like mean absolute error and R-squared value. The project provides actionable insights for coaches and analysts, aiding informed decisions on player selection and strategy. Its scalable methodology allows for integration of additional data sources, making it adaptable to other sports and laying the foundation for future research in sports analytics

2. SYSTEM ANALYSIS

2. SYSTEM ANALYSIS

SYSTEM ANALYSIS

In this project, system analysis involves examining the complexities of predicting cricket player performance by identifying key issues and asking, “What must be done to solve these challenges?” The analyst studies historical player statistics to determine relevant inputs for model development, focusing on feature selection to enhance accuracy. This process provides a clear understanding of the necessary steps for creating effective predictive models. Ultimately, the findings support informed decisions in player selection and strategy formulation, with potential applications extending to other sports.

2.1 PROBLEM DEFINITION

In the realm of cricket analytics, the application of machine learning techniques for predicting player performance represents a significant advancement in data-driven decision-making. However, effectively implementing a predictive model that optimizes accuracy poses a substantial challenge. Traditional evaluation methods often rely on static historical statistics, which may not capture the dynamic nature of player performance influenced by various factors. The core issue lies in the lack of a comprehensive and adaptable model that can accurately analyze and predict player performance in real-time. Without such a model, the ability to make informed decisions regarding player selection, strategy formulation, and performance improvement is severely constrained. Thus, the absence of a robust predictive mechanism is a critical barrier to maximizing the benefits of data analytics in cricket, highlighting the need for innovative approaches to address this urgent challenge.

2.2 EXISTING SYSTEM

In the realm of cricket analytics, traditional methods of assessing player accuracy often rely on manual evaluations and historical performance records. While these methods are informative, they may lack the precision and scalability offered by modern data-driven approaches. Additionally, the complexity of cricket dynamics—encompassing the interplay of batting, bowling, and fielding statistics—presents challenges in accurately gauging player accuracy through conventional means. Although some statistical models exist to analyze player performance, they often overlook the intricate relationships between various performance metrics and may struggle to adapt to evolving game dynamics. Moreover, the reliance on manual data processing and analysis can be time-consuming and prone to errors. Recognizing these limitations, there is a growing interest in leveraging machine learning techniques to enhance player accuracy prediction in cricket. By harnessing advanced algorithms capable of processing vast datasets and identifying intricate patterns, machine learning approaches offer the potential to revolutionize player assessment in cricket. This study aims to build upon existing methodologies by employing machine learning algorithms to develop accurate and scalable models for predicting player accuracy, thereby addressing the limitations of current manual and statistical approaches.

2.2.1 LIMITATIONS OF EXISTING SYSTEM

- Manual Evaluations
- Static Historical Data
- Complexity of Dynamics
- Inflexibility
- Time-Consuming Processes
- Error-Prone

2.3 PROPOSED SYSTEM

The proposed system aims to overcome the limitations of existing methods by leveraging machine learning techniques for the accurate prediction of player performance in cricket. By utilizing advanced algorithms capable of processing large volumes of player data, the system introduces objectivity into player assessment through data-driven analysis rather than subjective evaluations. It considers multiple performance metrics—such as batting average, bowling average, strike rate, economy rate, and fielding statistics—providing a comprehensive understanding of player contributions to the game. Additionally, the system offers scalability through automated data processing, enabling efficient evaluation across diverse datasets and players. By employing adaptable machine learning models, it ensures relevance and accuracy in predictions over time, facilitating real-time analysis that allows coaches, team selectors, and analysts to make timely, informed decisions based on up-to-date insights.

2.3.1 ADVANTAGES OF THE PROPOSED SYSTEM

- Objectivity
- Comprehensive Analysis
- Scalability
- Adaptability
- Real-Time Insights
- Improved Decision-Making

2.4 FEASIBILITY STUDY

2.4.1 ECONOMIC FEASIBILITY

The developing system must be justified by cost and benefit. It is important to focus efforts on projects that provide the best return as quickly as possible. One of the key factors influencing the development of a new system is the associated costs.

The following are some of the important financial questions considered during this study:

2.4.1.1 The costs involved in conducting a full system investigation.

2.4.1.2 The cost of hardware and software.

2.4.1.3 The benefits in terms of reduced costs or fewer costly errors.

Since the system is developed within a limited budget, there is no excessive expenditure for the proposed system. Most of the technologies used are freely available, with only customized products requiring purchase. This ensures that the system is economically feasible for development.

2.4.2 TECHNICAL FEASIBILITY

This study is carried out to assess the technical feasibility, specifically the technical requirements of the system. Any system developed should not place excessive demands on the available technical resources, as this would result in a burden on the client. The developed system should have modest requirements, ensuring that minimal or no changes are needed for its implementation.

2.4.3 SOCIAL FEASIBILITY

This includes the following questions:

2.2.3.1 Can machine learning techniques effectively predict the performance of cricket players?

2.2.3.2 Will the proposed system provide accurate and reliable forecasts?

The project would be beneficial because it meets the objectives of predicting cricket player performance when developed and implemented. All technical and predictive aspects have been carefully considered, and it is concluded that the project is technically feasible and effective in its purpose.

2.5 HARDWARE & SOFTWARE REQUIREMENTS

2.5.1 HARDWARE REQUIREMENTS:

Hardware interfaces specifies the logical characteristics of each interface between the software product and the hardware components of the system. The following are some hardware requirements:

- Processor : i3 or Above
- Hard disk : 20GB or Above.
- RAM : 4GB or Above.
- Monitor : 5 inches or above.

2.5.2 SOFTWARE REQUIREMENTS:

Software Requirements specifies the logical characteristics of each interface and software components of the system. The following are some software requirements:

- Operating system : Pentium IV or Above
- Languages : Python 3.7.0 Version
- Framework : Django

3. ARCHITECTURE

3. ARCHITECTURE

3.1 PROJECT ARCHITECTURE

This project architecture shows the procedure followed for predicting accuracy of players in the cricket using machine learning, starting from input to final prediction.

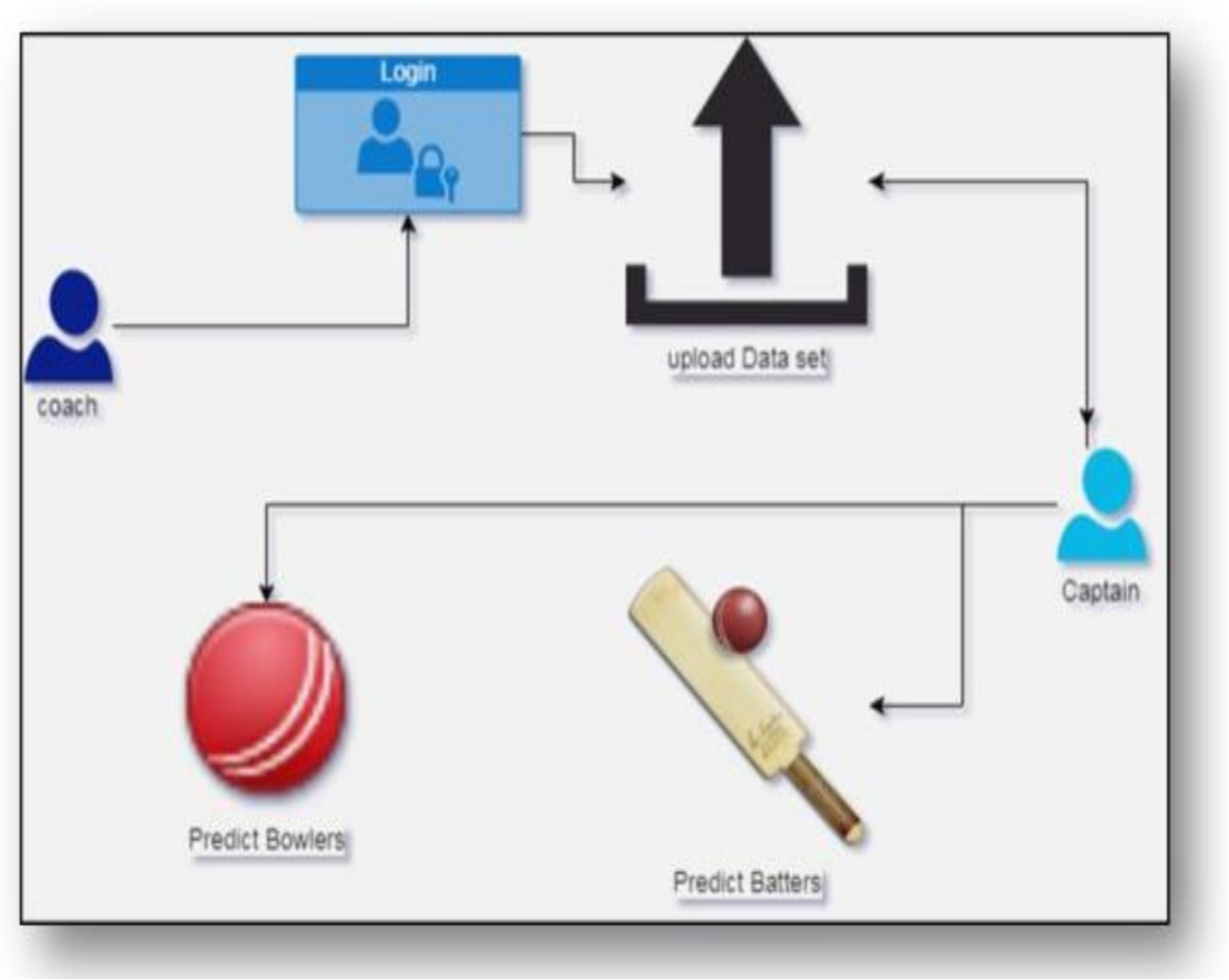


Figure 3.1: Project Architecture for Predicting Accuracy of Players in the Cricket

3.2 DESCRIPTION

Input Data

Input data is generally in CSV format, where player statistics and performance metrics are fetched and mapped into a structured data frame.

Reading Data

The Pandas library is used to read the data into the data frame for further processing and analysis.

Separating Features

In this step, we separate the features that will be used to train the model, assigning a target value (e.g., predicted performance) for the relevant features.

Normalization

Normalization is a crucial step when dealing with varying scales in features, as larger values can lead to higher computational costs and longer processing times. To enhance computational efficiency, we normalize the data values.

Training and Test Data

The training data is passed to the chosen machine learning algorithm to train the model, while the test data is used to evaluate the model's accuracy in making predictions.

Model Selection

The purpose of selecting a specific machine learning model for this project is based on its efficiency and accuracy, which have been validated through comparisons with other models. This allows us to effectively predict the performance of batters and bowlers, as well as support decision-making for coaches and captains.

3.3 USE CASE DIAGRAM

In the use case diagram, we have two actors: the user and the captain. The user has the ability to log in, predict the top 15 batsmen and bowlers, and log out. The captain can view the predicted top 15 batsmen and bowlers based on the data provided by the user.

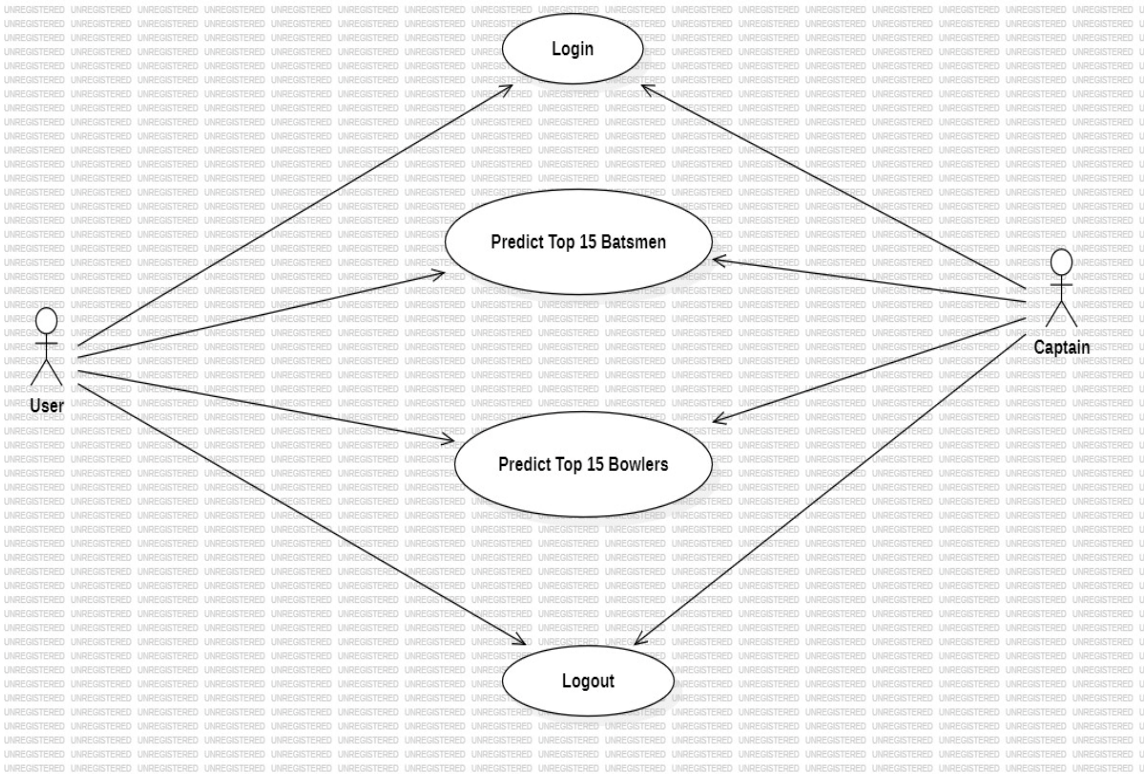


Figure 3.3: Use Case Diagram for predicting accuracy of cricket players

3.4 CLASS DIAGRAM

The diagram illustrates the relationship between the User and Captain classes. Both actors have similar functionalities like logging in, predicting the top 15 batsmen and bowlers, and logging out. The User class, however, contains additional attributes like contact information and user type.

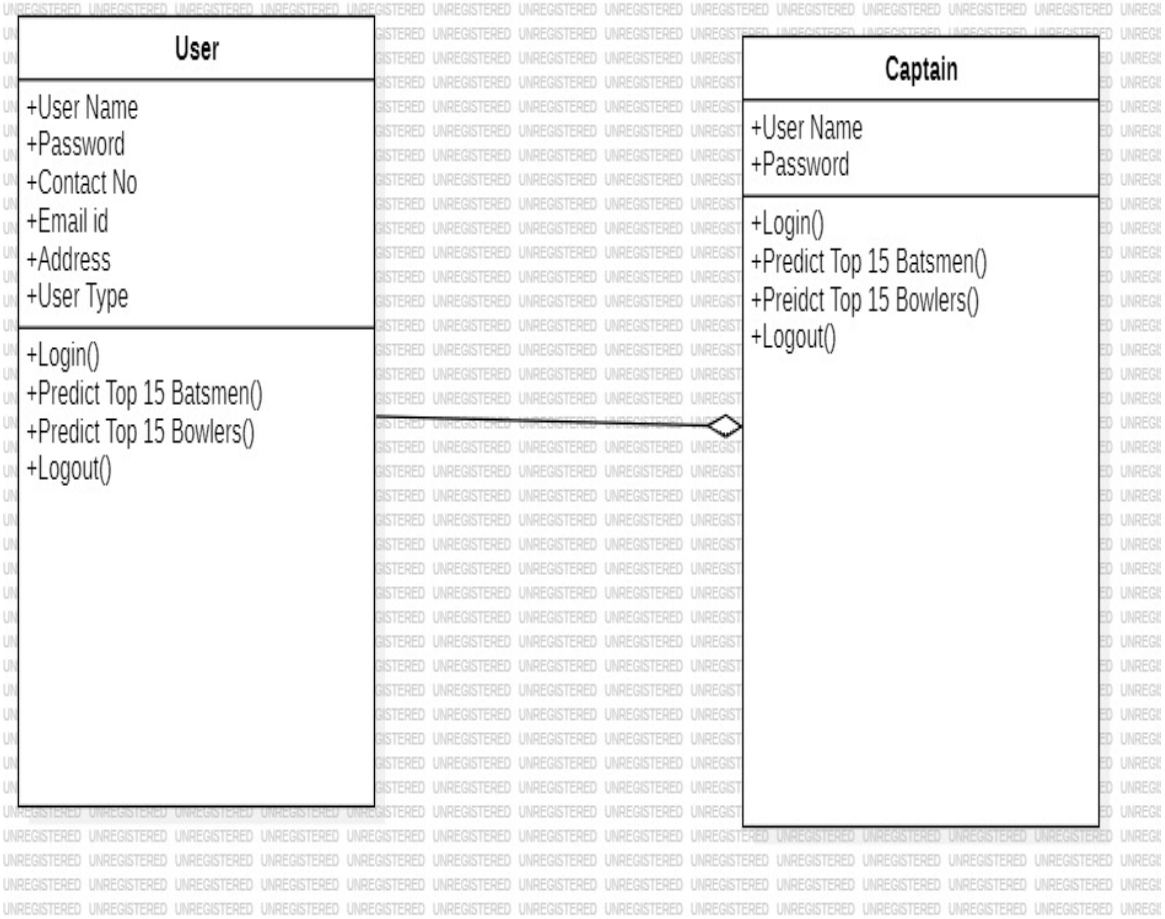


Figure 3.4: Class Diagram for predicting accuracy of cricket players

3.5 SEQUENCE DIAGRAM

The sequence diagram shows the interaction between the user, captain and database. It outlines the steps involved in logging in, predicting the top 15 batsmen and bowlers, and logging out, with communication between the system components at each step.

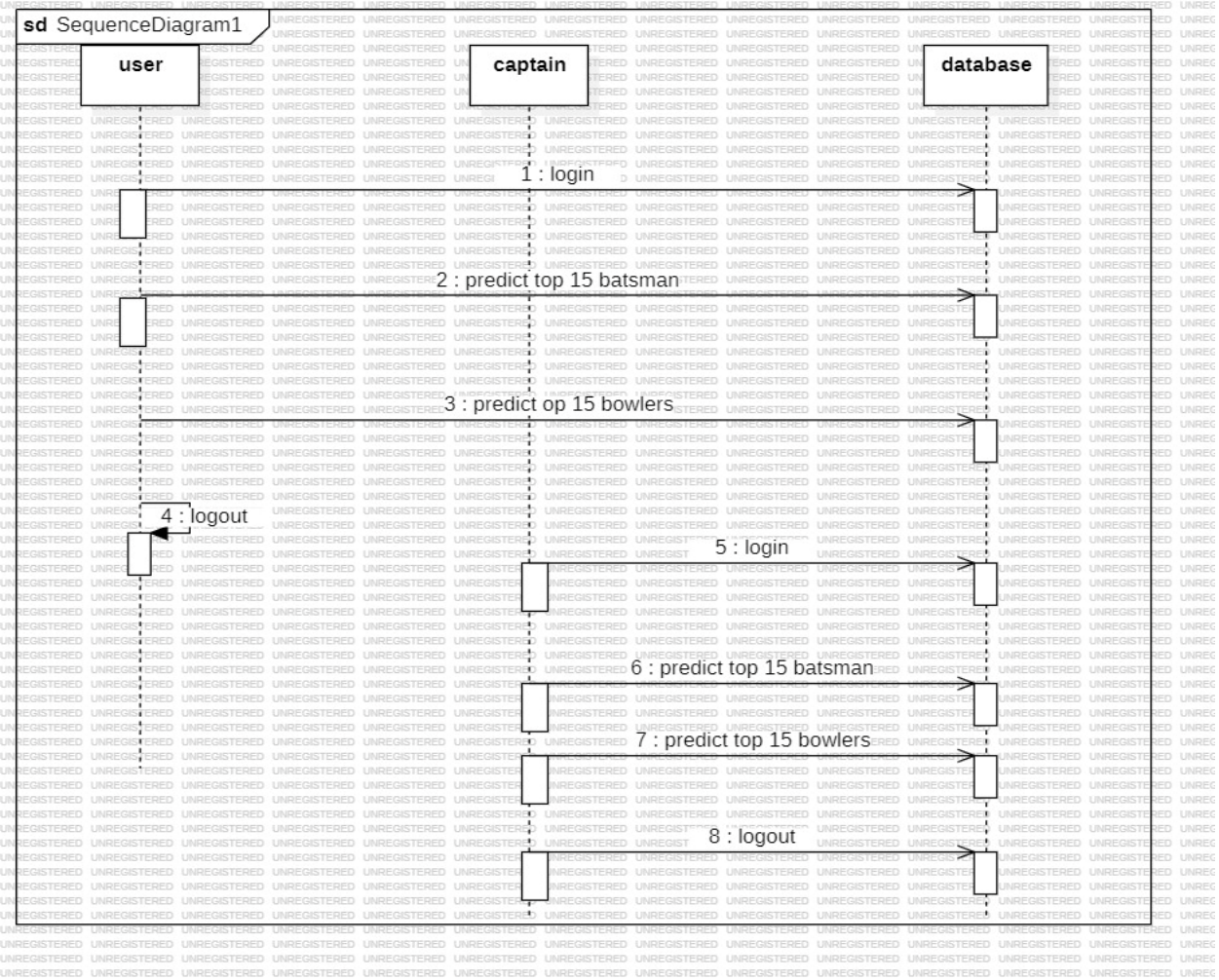


Figure 3.5: Sequence Diagram for predicting accuracy of cricket playe

3.6 ACTIVITY DIAGRAM

The activity diagram starts with Login, followed by Predicting Top 15 Batsmen and Bowlers based on data. After displaying the results, the process ends with Logout.

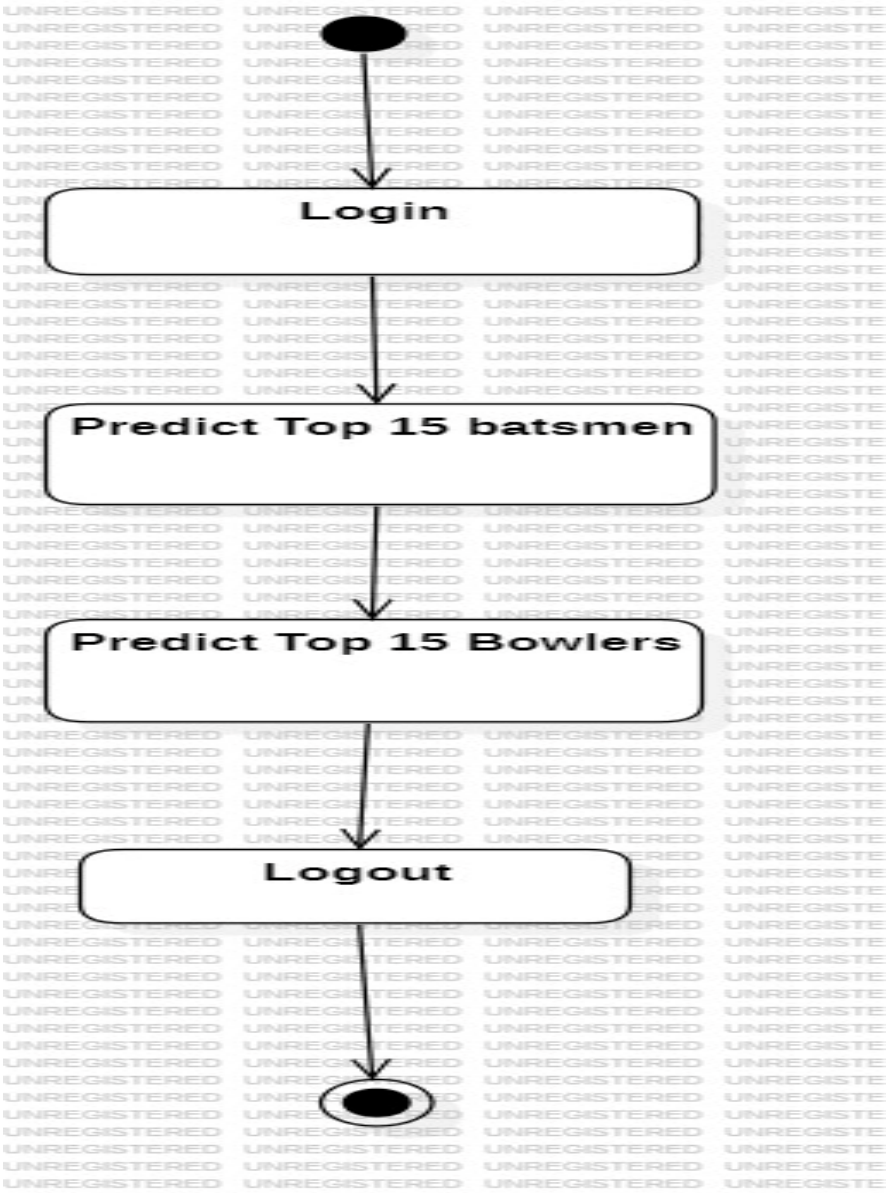


Figure 3.6: Activity Diagram for User for predicting accuracy of cricket players

4. IMPLEMENTATION

4. IMPLEMENTATION

4.1 SAMPLE CODE

Test.py

```
import pandas as pd
import numpy as np
from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import StandardScaler
from hmmlearn import hmm
import operator
import os
import pickle
'''
dataset = pd.read_csv("Dataset/bat.csv", usecols = ['name_x', 'runs_x', 'balls', 'strike_rate',
'fours', 'sixes', 'how_out', 'run_rate'])

le1 = LabelEncoder()
le2 = LabelEncoder()
scaler = StandardScaler()

dataset['name_x'] = pd.Series(le1.fit_transform(dataset['name_x'].astype(str)))
dataset['how_out'] = pd.Series(le2.fit_transform(dataset['how_out'].astype(str)))
dataset = dataset.values

X = dataset[:,0:dataset.shape[1]-1]
Y = dataset[:,dataset.shape[1]-1]

X = scaler.fit_transform(X)
print(X.shape)
print(Y.shape)
if os.path.exists("model/bat"):
    with open('model/bat', 'rb') as file:
        model = pickle.load(file)
```

```
    file.close()
else:
    model = hmm.GaussianHMM(10, "full", n_iter=500)
    model.fit(X)
    with open('model/bat', 'wb') as file:
        pickle.dump(model, file)
    file.close()
testData = pd.read_csv("Dataset/test_bat.csv", usecols = ['name_x', 'runs_x', 'balls',
'strike_rate', 'fours', 'sixes', 'how_out'])
player = testData['name_x']
testData['name_x'] = pd.Series(le1.transform(testData['name_x'].astype(str)))
testData['how_out'] = pd.Series(le2.transform(testData['how_out'].astype(str)))
testData = testData.values
X = scaler.transform(testData)
performance = []
predict = model.predict(X)
print(model.score(X))
selected = []
for i in range(len(predict)):
    if player[i] not in selected:
        selected.append(player[i])
        performance.append([player[i], predict[i]])
performance.sort(key = operator.itemgetter(1), reverse = True)
for i in range(0,15):
    print(performance[i])
dataset = pd.read_csv("Dataset/ball.csv", usecols = ['name_x', 'run_conceded', 'maidens',
'wickets', 'overs', 'economy', 'wides', 'no_balls', 'fours',
                                        'sixes', 'zeros', 'runs', 'over', 'run_rate'])
dataset.fillna(0, inplace = True)
le3 = LabelEncoder()
scaler1 = StandardScaler()
dataset['name_x'] = pd.Series(le3.fit_transform(dataset['name_x'].astype(str)))
X = dataset.values
```

```
X = scaler1.fit_transform(X)
print(X.shape)
if os.path.exists("model/ball"):
    with open('model/ball', 'rb') as file:
        model = pickle.load(file)
    file.close()
else:
    model = hmm.GaussianHMM(10, "full", n_iter=5000)
    model.fit(X)
    with open('model/ball', 'wb') as file:
        pickle.dump(model, file)
    file.close()
testData = pd.read_csv("Dataset/test_ball.csv", usecols = ['name_x', 'run_conceded',
'maidens', 'wickets', 'overs', 'economy', 'wides', 'no_balls', 'fours',
                                        'sixes', 'zeros', 'runs', 'over', 'run_rate'])
testData.fillna(0, inplace = True)
player = testData['name_x']
testData['name_x'] = pd.Series(le3.transform(testData['name_x'].astype(str)))
testData = testData.values
X = scaler1.transform(testData)
performance = []
predict = model.predict(X)
print(model.score(X))
selected = []
for i in range(len(predict)):
    if player[i] not in selected:
        selected.append(player[i])
        performance.append([player[i], predict[i]])
performance.sort(key = operator.itemgetter(1), reverse = True)
for i in range(0,15):
    print(performance[i])
```

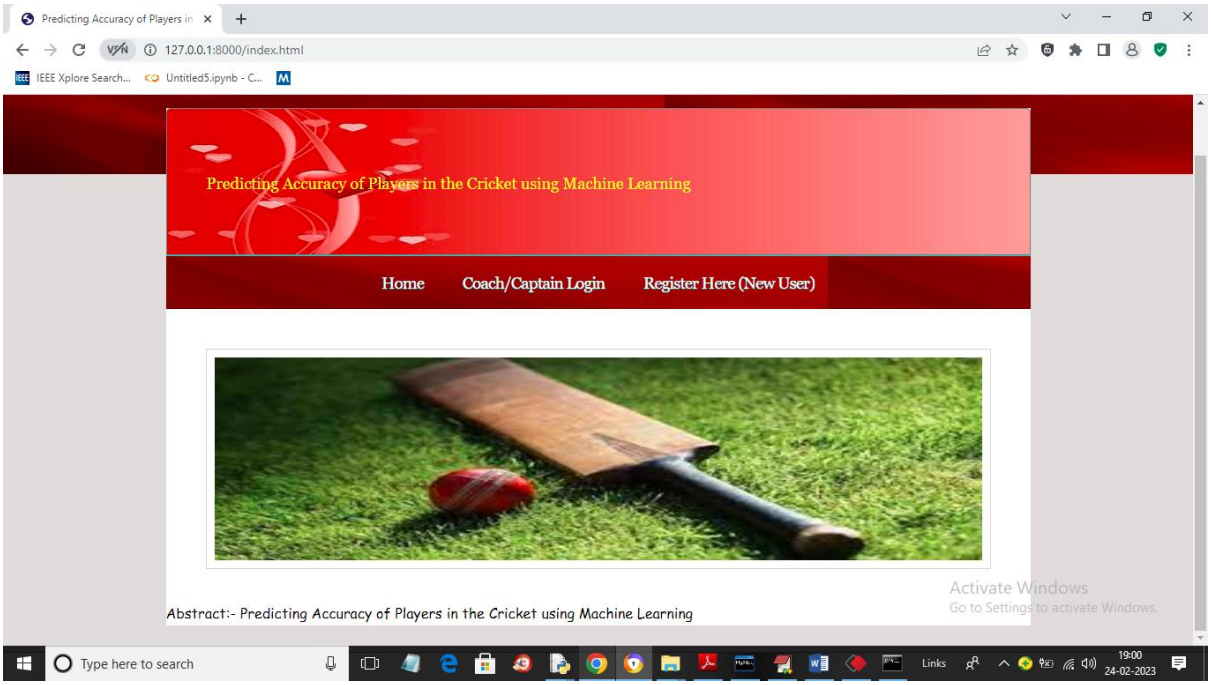
Manage.py

```
 #!/usr/bin/env python
import os
import sys

if __name__ == '__main__':
    os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'Cricket.settings')
    try:
        from django.core.management import execute_from_command_line
    except ImportError as exc:
        raise ImportError(
            "Couldn't import Django. Are you sure it's installed and "
            "available on your PYTHONPATH environment variable? Did you "
            "forget to activate a virtual environment?"
        ) from exc
    execute_from_command_line(sys.argv)
```

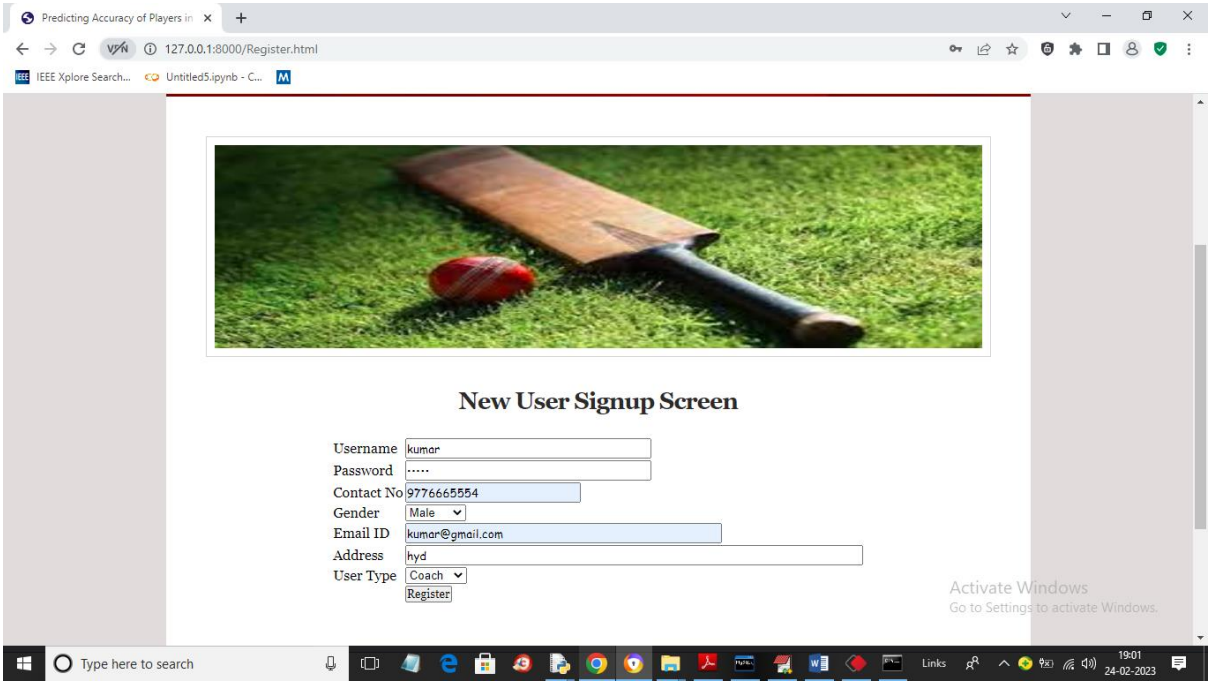
5. SCREENSHOTS

5.1 HOME PAGE



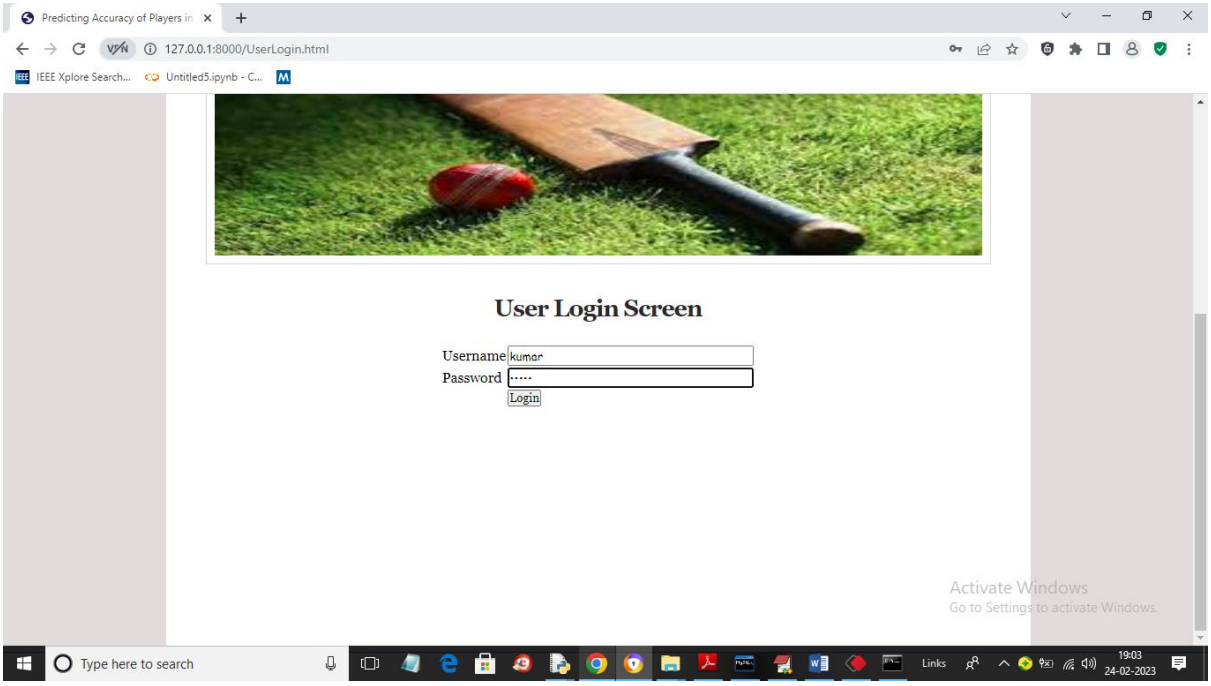
Screenshot 5.1: Home Page of predicting accuracy of cricket players

5.2 REGISTER (SIGN UP) PAGE



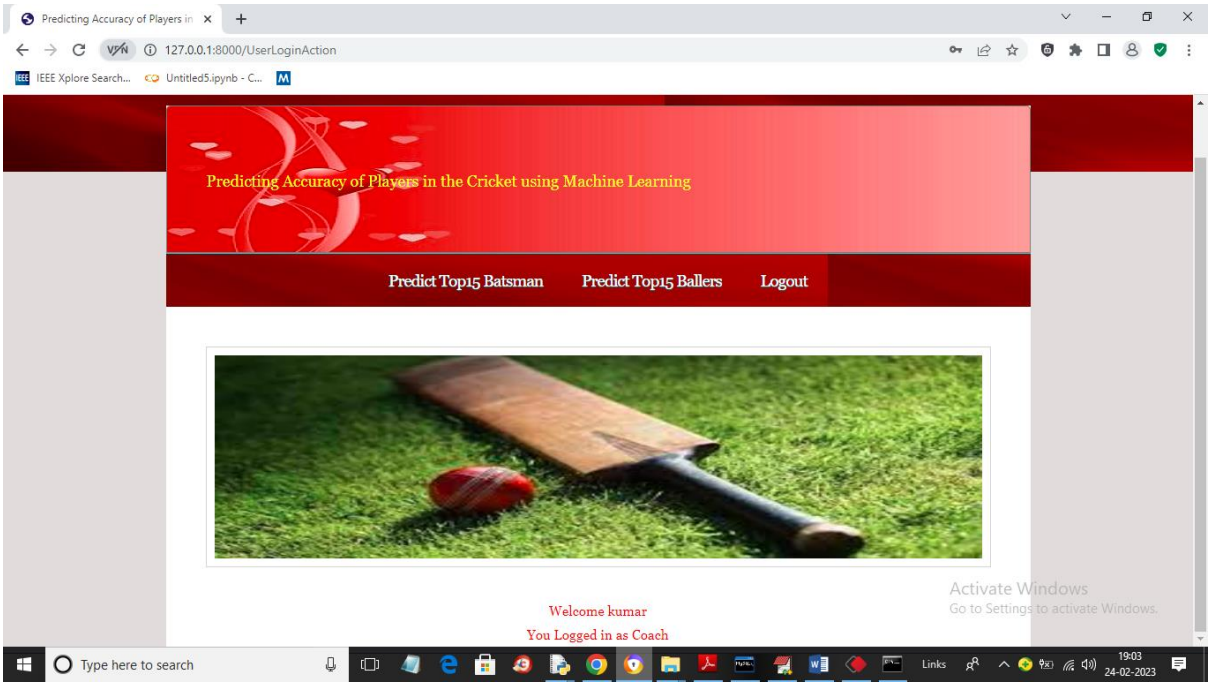
Screenshot 5.2: Sign Up Page of predicting accuracy of cricket players

5.3 USER LOGIN PAGE



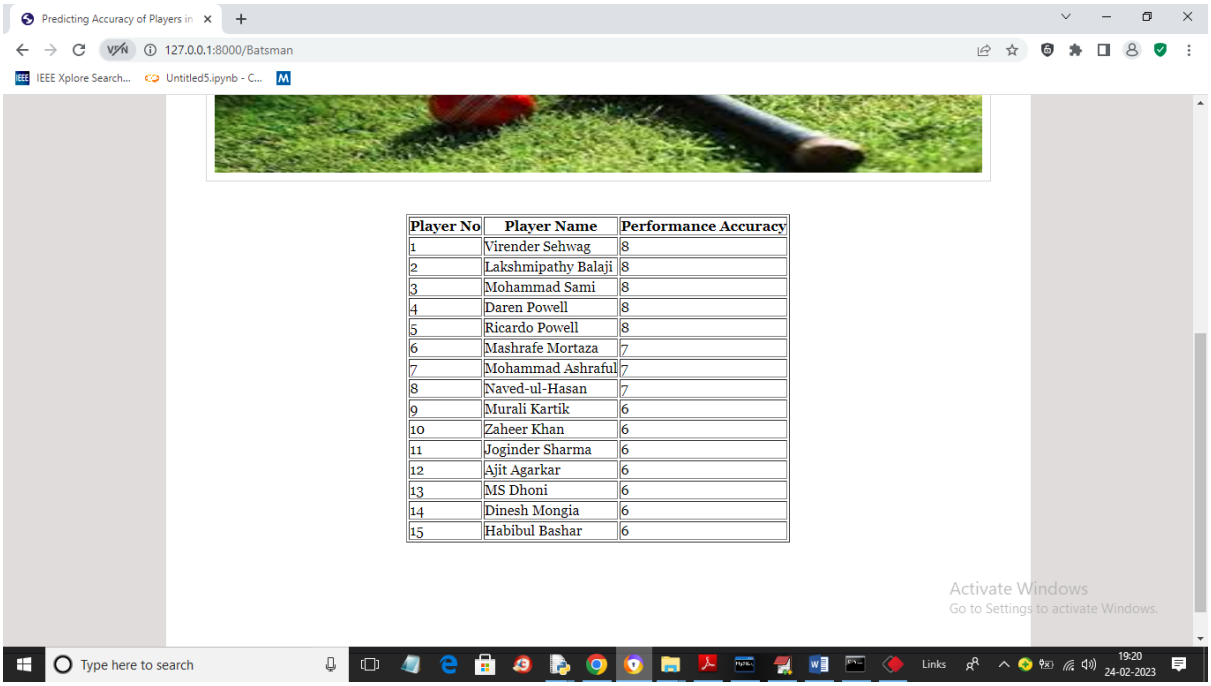
Screenshot 5.3: User Login Page of predicting accuracy of cricket players

5.4 RESULT PAGE



Screenshot 5.4: Result Page of predicting accuracy of cricket players

5.5 TOP 15 BATSMEN RESULT



Screenshot 5.5: Top 15 Batsmen result of predicting accuracy of cricket players

5.6 TOP 15 BOWLERS RESULT

Screenshot 5.6: Top 15 Bowlers result of predicting accuracy of cricket players

6. TESTING

6. TESTING

6.1 INTRODUCTION TO TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discoverevery conceivable fault or weakness in a work product. It provides a way to check the functionality of components, subassemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

6.2 TYPES OF TESTING

6.2.1 UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

6.2.2 INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

6.2.3 FUNCTIONAL TESTING

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Inp : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes.

6.3 TEST CASES

6.3.1 UPLOADING DATASET

Test case ID	Test case name	Purpose	Test Case	Output
1	User uploads batsmen dataset	Use it for identification	The user uploads the batsmen data	Uploaded successfully
2	User uploads bowlers dataset	Use it for identification	The user uploads the bowlers data	Uploaded successfully

6.3.2 CLASSIFICATION

Test case ID	Test case name	Purpose	Input	Output
1	Classification test 1	To check if the classifier performs its task	Batsmen dataset is given	Batsmen are predicted.
2	Classification test 2	To check if the classifier performs its task	Bowlers dataset is given	Bowlers are predicted

7. CONCLUSION

7. CONCLUSION & FUTURE SCOPE

7.1 PROJECT CONCLUSION

The project titled "Predicting the Performance and Accuracy of Cricket Players" is a machine learning-based application. This software utilizes historical cricket player data to forecast their future performance. It analyzes various skills such as batting, bowling, and fielding. The dataset used includes statistics like batting average, bowling average, strike rate, economy rate, and fielding metrics.

The system employs multiple machine learning algorithms such as linear regression, decision trees, random forests, and support vector machines to build predictive models. Feature selection techniques are used to identify key attributes for accurate prediction. Each model's performance is assessed using evaluation metrics like mean absolute error, mean squared error, and R-squared value.

The project successfully demonstrates the effectiveness of this approach in predicting the accuracy of cricket players. This system provides valuable insights for coaches, team selectors, and analysts, aiding in player selection, strategy formulation, and performance enhancement. Furthermore, this methodology can be extended to other sports for similar performance predictions.

7.2 FUTURE SCOPE

In the future, we can integrate additional machine learning algorithms, including convolutional neural networks, by downloading the necessary modules directly into the project files. The software can be further expanded to include various other modules, as the proposed system is developed with a future-oriented approach. We can also connect the system to additional databases, enhancing the model's ability to analyze more extensive data and improve the accuracy of performance predictions for cricket players.

8. BIBLIOGRAPHY

8. BIBLIOGRAPHY

8.1 REFERENCES

[1] Patel, M., & Raj, R. K. (2020). Predictive Analysis of Cricket Data Using Machine Learning Techniques. International Journal of Computer Sciences and Engineering, 8(11), 246-250.

[2] Shrivastava, A., & Rathi, V. (2019). Cricket Analytics using Machine Learning. International Journal of Computer Applications, 975(15), 8887-8890.

[3] Sharma, N., & Bhola, S. (2018). Predicting Cricket Match Outcomes Using Machine Learning Techniques. International Journal of Computer Sciences and Engineering, 6(9), 35-38.

[4] Khan, N., Siddique, M., & Ahmed, A. (2018). Predicting the Outcome of a Cricket Match Using Machine Learning. International Journal of Advanced Computer Science and Applications, 9(10), 485-491.

[5] Mubin, O., & Sharma, S. (2020). Machine Learning Application in Cricket Analytics. International Journal of Computer Science and Information Security, 18(3), 72-76.

[6] Kumar, R., & Vashisth, N. (2019). Cricket Prediction Using Machine Learning. International Journal of Computer Sciences and Engineering, 7(5), 454-459.

[7] Vashisht, P., & Mathur, A. (2018). Predictive Modeling in Cricket using Machine Learning Algorithms. International Journal of Computer Sciences and Engineering, 6(10), 134-140.

[8] Sharma, N., & Bhola, S. (2019). Predicting Cricket Match Outcome Using Machine Learning. International Journal of Computer Sciences and Engineering, 7(7), 20-24.

[9] Nair, A., & Rajendran, R. (2017). Predicting Cricket Match Outcome Using Machine Learning Techniques. International Journal of Engineering and Technology, 9(4), 3165-3169.

[10] Singh, P., & Kumar, A. (2019). Predictive Analysis of Cricket using Machine Learning Algorithms. International Journal of Computer Applications, 182(23), 23-27.

8.2 WEBSITES

- https://github.com/Nazeer-786/MiniProject/tree/main/MINIPROJ