s

**IPL SCORE PREDICTION SYSTEM**

**A Project Report**

Submitted in partial fulfillment of the

Requirements for the award of the Degree of

**BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)**

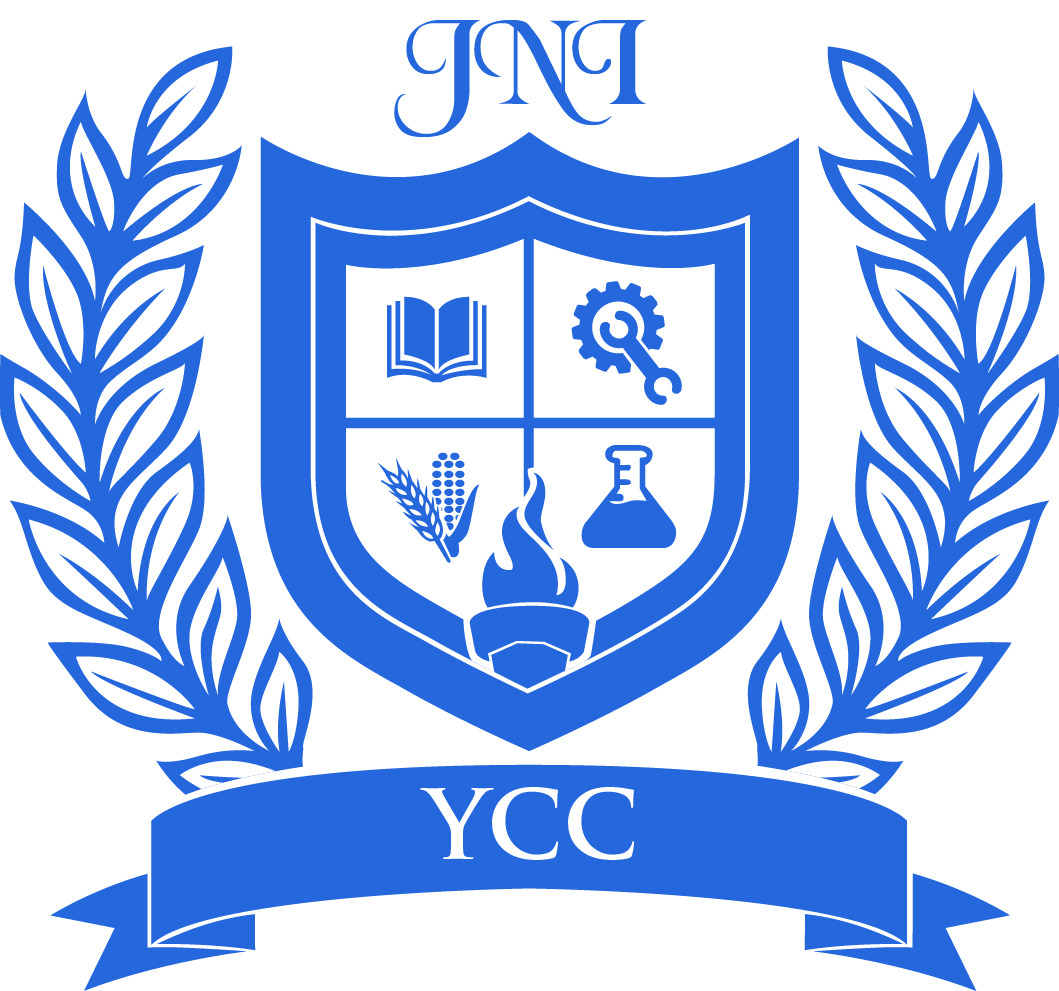
**By**

**Pratik Shinde**

seat no:

**under the esteemed guidance of**

**Prof. Sarita Kumari**

****

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**YASHWANTRAO CHAVAN COLLEGE (YCC)**

***(Affiliated to University of Mumbai)***

**NAVI MUMBAI,** 400709

**MAHARASHTRA**

**2021-22**

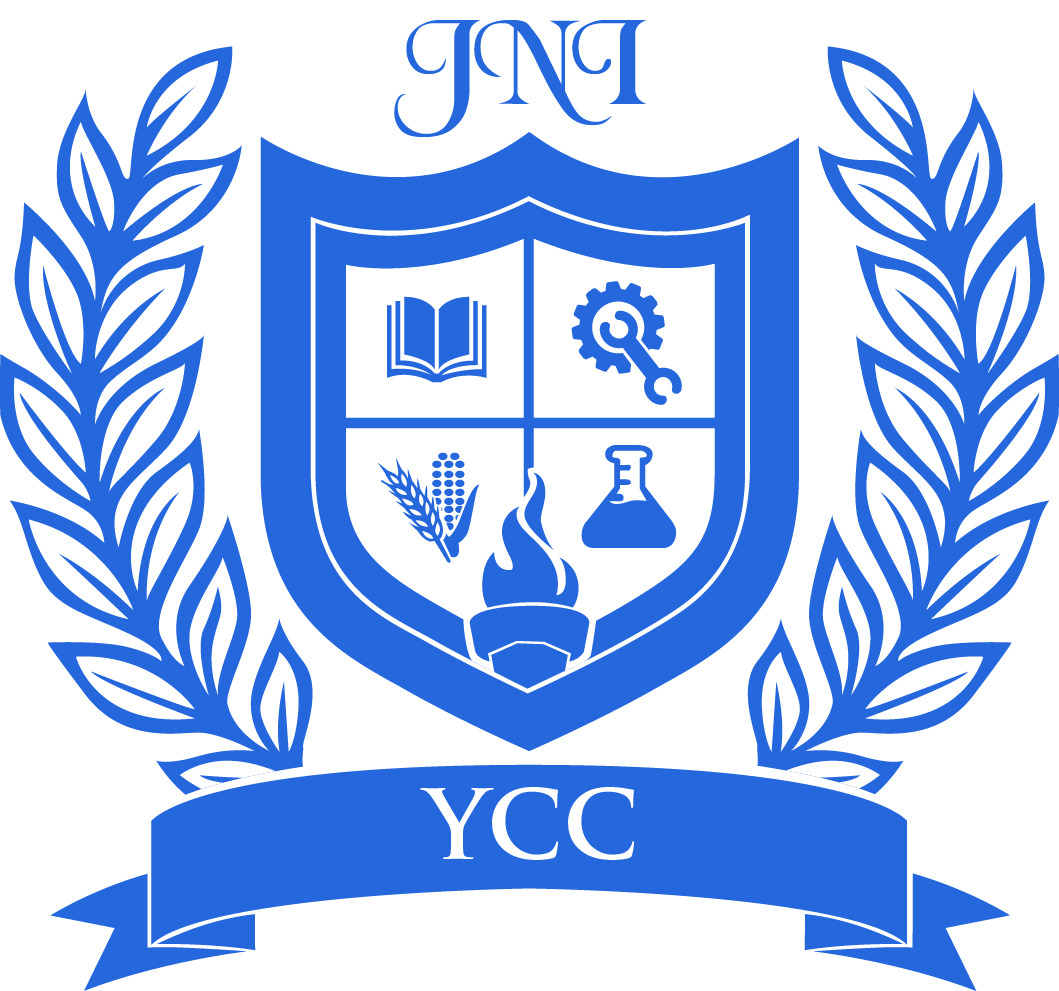
**2021-22**

**YASHWANTRAO CHAVAN COLLEGE (YCC)**

***(Affiliated to University of Mumbai)***

**NAVI MUMBAI-MAHARASHTRA-**400709

**DEPARTMENT OF INFORMATION TECHNOLOGY**



**CERTIFICATE**

This is to certify that the project entitled, “IPL Score Prediction System”, is bonafied work of Pratik Shinde bearing seat no: submitted in partial fulfillment of the requirements for the award of degree of BACHELOR OF SCIENCE in INFORMATIONTECHNOLOGY from University of Mumbai.

**Internal Guide Head of Department**

**External Examiner Principle**

**Date: College Seal**

**ABSTRACT**

This paper is about a model that can predict the projected score of 1st inning as well as the winner in a IPL cricket match. The performance of model depends on various features like wickets taken in last 5 overs, runs scored in last 5 overs, overs, overall score and wickets at current ball. The proposed model contains data from IPL matches played between years 2008 and 2019. This paper will give us step by step insights on how one can predict projected

score of 1st inning while the match is still in progress. Linear Regression algorithm is used to predict the score. This model explains about 75.226% of data. The model specifically emphasize on using the data from past 5 overs to predict what might be the projected score of the match which has not been considered in any existing model. Using this model, we can get good insight during the match on how much score will the current batting team obtain.

**Keywords**: Analysis, Efficiency, Evaluation, Cricket, First innings.

**ACKNOWLEDGEMENT**

The success and final outcome of this project required a lot of guidance and assistance from many people and I am extremely privileged to have got this all along the completion of my project. All that I have done is only due to such supervision and assistance and I would not forget to thank them.

I respect and thank to our Principal Dr. Laxman Talinekar, for providing me an opportunity to do the project work in YASHWANTRAOCHAVANCOLLEGE **(**YCC**)** and giving us all support and guidance, which made me complete the project duly. I am extremely thankful to her for providing such a nice support and guidance, although she had busy schedule managing the corporate affairs.

I owe my deep gratitude to our project guide Dr.Triveni Kaul, Prof. SaritaKumari ,Prof. Pushpanjali Patil.who took keen interest on our project work and guided us all along, till the completion of our project work by providing all the necessary information for developing a good system.

I am thankful to and fortunate enough to g et constant encouragement, support and guidance from all Teaching staffs of Computer Science department which helped us in successfully completing our project work. Also, I would like to extend our sincere esteems to all staff in laboratory for their timely support.

**DECLARATION**

I hereby declare that the Project entitled, “IPL SCORE PREDICTION SYSTEM” done at YASHWANTRAOCHAVANCOLLEGE **(**YCC**),** has not been in any case duplicated to submit to any other university for the award of any degree. To the best of my knowledge other than me, no one has submitted to any other university.

The Project is done in partial fulfillment of the requirements for the award of degree of BACHELOR OF SCIENCE (INFORMATIONTECHNOLOGY) to be submitted as final semester project as part of our curriculum.

Pratik Shinde

Name and Signature of the Student

|  |  |  |
| --- | --- | --- |
| **SR.NO** | **TOPIC** | **PAGE.NO** |
| 1 | **INTRODUCTION** | 1-5 |
| [1.1 Background Of Study](#__RefHeading___Toc2996_1360538424)  [1.2 Machine Learning In Cricket](#__RefHeading___Toc2998_1360538424)  [1.3 Indian Premier League](#__RefHeading___Toc3000_1360538424)  [1.4 Objective](#__RefHeading___Toc3002_1360538424)  [1.5 Literature Review](#__RefHeading___Toc3004_1360538424)  [1.6 Scope](#__RefHeading___Toc3006_1360538424)  [1.7 Problem Statement](#__RefHeading___Toc3008_1360538424) |
| 2 | [**Survey of Technologies**](#__RefHeading___Toc3018_1360538424) | 6-13 |
| [2.1 HTML](#__RefHeading___Toc3010_1360538424)  [2.2 CSS](#__RefHeading___Toc3012_1360538424)  [2.3 JAVASCRIPT](#__RefHeading___Toc3014_1360538424)  [2.4 PYTHON](#__RefHeading___Toc3016_1360538424)  [2.5 Jupyter Notebook](#__RefHeading___Toc3016_1360538424)  [2.6 Deployment](#__RefHeading___Toc3016_1360538424) |
| 3 | [**System Analysis**](#__RefHeading___Toc3020_1360538424) | 14-24 |
| [3.1 PROPOSED SYSTEM](#__RefHeading___Toc3022_1360538424)  [3.1.1 ALGORITHMS & TECHNIQUES](#__RefHeading___Toc3024_1360538424)  [3.2 REQUIREMENT ANALYSIS](#__RefHeading___Toc3026_1360538424)  [3.2.1 HARDWARE REQUIREMENTS](#__RefHeading___Toc3028_1360538424)  [3.2.2 SOFTWARE REQUIREMENTS](#__RefHeading___Toc3030_1360538424)  [3.3 PLANNING AND SCHEDULING](#__RefHeading___Toc3032_1360538424)  [3.4 CONCEPTUAL MODEL](#__RefHeading___Toc3034_1360538424)  [3.4.1 Activity Diagram](#__RefHeading___Toc3036_1360538424)  [3.4.2 Use Case Diagram](#__RefHeading___Toc3038_1360538424) |
| 4 | [**System Design**](#__RefHeading___Toc3040_1360538424) | 25-32 |
| [4.1 MODULE DIVISION](#__RefHeading___Toc3042_1360538424)  [4.2 PROCEDURAL DESIGN](#__RefHeading___Toc3044_1360538424)  [4.2.1 DATA FLOW DIAGRAM](#__RefHeading___Toc3046_1360538424)  [4.2.2 ALGORITHM DESIGN](#__RefHeading___Toc3048_1360538424)  [4.3 USER INTERFACE DESIGN](#__RefHeading___Toc3050_1360538424)  [4.4TEST CASE DESIGN](#__RefHeading___Toc3052_1360538424) |
| 5 | **Implementation and Testing** | 33-41 |
| [5.1 IMPLEMENTATION APPROACHES](#__RefHeading___Toc3056_1360538424)  [5.2 CODE DETAILS AND EFFICIENCY](#__RefHeading___Toc3058_1360538424)  [5.3 TESTING APPROACH](#__RefHeading___Toc3060_1360538424)  [5.4 MODIFICATIONS AND IMPROVEMENTS](#__RefHeading___Toc3062_1360538424) |
| 7 | **Conclusion** | 42 |
| 8 | [Referenc](#__RefHeading___Toc3066_1360538424)e | 43 |
| [7.1 REFERENCE](#__RefHeading___Toc3068_1360538424)  7.2 BIBLIOGRAPHY |
|  | **LIST OF FIGURES** |  |
| [Fig. 1 Yearly Match Analysis using Matplotlib & Seaborn](#_Toc98760188) 9  [Fig. 2 Creating Dataframe using Pandas](https://d.docs.live.net/a4068e3cc157d41b/Documents/FinalYearProject2.docx" \l "_Toc98760189) 10  [Fig. 3 Proposed System 1](#_Toc98760190)5  [Fig. 4 Linear Regression](https://d.docs.live.net/a4068e3cc157d41b/Documents/FinalYearProject2.docx" \l "_Toc98760191) 16  [Fig. 5 Multilayer Perception Neural Network](https://d.docs.live.net/a4068e3cc157d41b/Documents/FinalYearProject2.docx" \l "_Toc98760192) 17  [Fig. 6 Activity Diagram](#_Toc98760193) 22  [Fig. 7 Use Case Diagram](#_Toc98760194) 23  [Fig. 8 Data Flow Diagram 2](#_Toc98760195)7  [Fig. 9 Home Page 2](#_Toc98760196)9  [Fig. 10 Result Page 2](#_Toc98760197)9  [Fig. 11 Validation same team Page](#_Toc98760197) 30  [Fig. 12 Wi*ckets validation Page*](#_Toc98760197) 30  [Fig. 1](https://d.docs.live.net/a4068e3cc157d41b/Documents/FinalYearProject2.docx" \l "_Toc98760198)3 Beta Testing 39 |
|  | **LIST OF TABLES** |  |
| 1. Gnatt Chart  2. Test Case  3 Beta Testing |

# **Chapter 1:**

# **Introduction**

## **Background Of Study**

With technology growing more and more advanced in the last few years, an in-depth acquisition of data has become relatively easy. As a result, Machine Learning is becoming quite a trend in sports analytics because of the availability of live as well as historical data. Sports analytics is the process of collecting past matches data and analyzing them to extract the essential knowledge out of it, with a hope that it facilitates in effective decision making. Decision making may be anything including which player to buy during an auction, which player to set on the field for tomorrow’s match, or something more strategic task like, building the tactics for forthcoming matches based on players’ previous performances.

Machine Learning can be used effectively over various occasions in sports, both on-the-field and off-the-field. When it is about on-the-field, machine learning applies to the analysis of a players fitness level, design of offensive tactics, or decide shot selection. It is also used in predicting the performance of a player or a team, or the outcome of a match. On the other hand, the off-the-field scenario concerns the business perspective of the sport, which includes understanding sales pattern (tickets, merchandise) and assigning prices accordingly. The main focus is the healthy growth in business and profitability of the team owners and other stakeholders. On-the-field analytics generally make use of supervised machine learning algorithms, example: (i) regression for calculating the fitness of a player, (ii) classification for predicting an outcome of a match; while off-the-field analytics concerns around performing sentiment analysis to understand peoples opinion about a player or a team or a sport league. At present, Twitter has become one of the primary sources of data for sentiment analysis.

## **Machine Learning In Cricket**

In cricket, to predict an outcome of a match, the primary task is to extract out the essentials factors (features) which affect result of a match. Interesting works have been done in the field of predicting outcome in cricket. The literature survey concluded that the majority of the published works which predicted a result of a cricket match prior were for the test or one-day international cricket format. Analysts has analyzed the factors like home field advantage, winning the toss, game plan (first batting or first fielding) and the effect of for one-day cricket format. They have mentioned in their work that in one-day cricket format, home ground advantage, past performances, venue, performance against the specific opposition, current form are statistically significant in predicting total runs and predicting the outcome of a match. Similarly other analysts, mathematical calculators discusses modelling home-runs and nonhome runs prediction algorithms and considers taking runs, wickets, frequency of being all-out as historical features into their prediction model. But, they have not leveraged bowlers features and have given more emphasis to batsmen. They have proposed a tool that predicts match score, but player performance has not been considered into their model.

## **Indian Premier League**

Introduction Indian Premier League (IPL) is a professional cricket league based on Twenty20 format and is governed by Board of Control for Cricket in India. The league happens every year with participating teams name representing various cities of India. There are many countries active in organizing Twenty20 cricket leagues. While most of the leagues are being overhyped and team franchises are routinely losing money, IPL has stood out as an exception. As reported by ESPNcricinfo, with Star Sports spending $2.5 billion for exclusive broadcasting rights, the latest season of IPL (2018, 11th) saw 29% increment in the number of viewers including both the digital streaming media and television. The 10th season had 130 million people streaming the league through their digital devices and 410 million people watching directly on the TV. The numbers prove that IPL is a successful Twenty20 format based cricket league.

## **Objective**

Once the model has generated scores for all IPL players, we choose a team’s best playing XI using an algorithm and add all the points of the best XI players to get the total team score. This algorithm then generates the winner of two teams using the team evaluation points out that the machine learning algorithm predicted.

Data-driven decisions make the difference between keeping up with the competition or bowing out of the competition. The growth of machine learning has been propelled by the immense amount of available data, affordable data storage, and more powerful and less expensive data processing.

## **Literature Review**

1. Describes about significant challenges that we face for accurate prediction including the various parameters which affect the outcome of the match. The ball movement gets changed from every over, so it is considered being important to predicting the outcome of each match on every ball. Here they had developed a model that predicts the match result of every ball played.
2. Explains about the concept of identifying rising stars in cricket domain using some techniques. Rising stars can be predicted by both bats as well as bowling teams. Distinct features like concept of co-players, team and opposite teams are presented with their mathematical formulation.
3. Explained the outcome of ODI match depends on various factors. The list of key features is home-field advantages, winning the toss, game plan, venue and season. Linear Regression, SVM are the different types of algorithm used for model building. Linear Regression is applied for data that had been already obtained from previous matches. SVM used for predictive analysis. It was found that SVM was proved to be a better model based on both the parameters used to predict accuracy and model outcome.
4. Proposed a model using multiple variable linear regression and logistic regression to predict the score in different innings and also the winner of the match using random forest algorithm. Came up with live cricket score prediction using linear regression and naïve bayes classifier.
5. Proposed a new methodology for analyzing the error of classifiers and model selection measures to analyze the decision tree algorithm.
6. Proposed a model used matrix factorization technique to analyze and predict the winner in ODI cricket match.
7. Proposed a solution to calculate the weight age of a team based on players’ past performance of IPL using linear regression.

## **Scope**

IPL has over a billion viewers and everyday people watch matches of hours on television or live and generate billions of views. The main objective of this project is to focus on how data can be analysed visually to predict the scores of the teams, which will help them to idealize the final score of 1st inning.

Companies can use this project to understand how effective and penetrative their auction programs are. This can tell the companies when is the slow period or spike in performance and attribute the same to certain marketing campaign. Applications for IPL data can be endless. For example, Companies can analyze how much to spend on each player during auction.

## **Problem Statement**

In this analysis of IPL matches from 2008 to 2017 is done using python packages like pandas, matplotlib and seaborn. This Exploratory Data Analysis will help us to find patterns in data, determining relationships in data. We will try to identify the team that has more chance to win the upcoming seasons by finding observations like success rate of each team, identify the team that has won maximum seasons, best defending and chasing team and analyse the toss decisions etc.

**Chapter 2:**

**Survey of Technologies**

## **2.1 HTML**

The HyperText Markup Language, or HTML is the standard markup language for documents designed to be displayed in a [web browser](https://en.wikipedia.org/wiki/Web_browser). It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as [JavaScript](https://en.wikipedia.org/wiki/JavaScript).

[HTML elements](https://en.wikipedia.org/wiki/HTML_element) are the building blocks of HTML pages. With HTML constructs, [images](element) and other objects such as [interactive forms](https://en.wikipedia.org/wiki/Fieldset) may be embedded into the rendered page. HTML provides a means to create [structured documents](https://en.wikipedia.org/wiki/Structured_document) by denoting structural [semantics](https://en.wikipedia.org/wiki/Semantics) for text such as headings, paragraphs, lists, l[inks](https://en.wikipedia.org/wiki/Hyperlink), quotes and other items. HTML elements are delineated by tags, written using [angle brackets](Bracket). Tags such as<img /> and <input /> directly introduce content into the page. Other tags such as <p>surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

## **2.2 CSS**

Cascading Style Sheets(CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as [HTML](https://en.wikipedia.org/wiki/HTML). CSS is a cornerstone technology of the [World Wide Web](https://en.wikipedia.org/wiki/World_Wide_Web), alongside HTML and [JavaScript](https://en.wikipedia.org/wiki/JavaScript).

CSS is designed to enable the separation of presentation and content, including [layout](https://en.wikipedia.org/wiki/Page_layout), [colors](https://en.wikipedia.org/wiki/Color), and [fonts](https://en.wikipedia.org/wiki/Typeface). This separation can improve content [accessibility](https://en.wikipedia.org/wiki/Accessibility), provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file which reduces complexity and repetition in the structural content as well as enabling the .css file to be [cached](https://en.wikipedia.org/wiki/Cache_(computing)) to improve the page load speed between the pages that share the file and its formatting.

## **2.3 JAVASCRIPT**

JavaScript often abbreviated as JS, is a programming language that conforms to the [ECMAScript](https://en.wikipedia.org/wiki/ECMAScript)specification.JavaScript is [high-level](https://en.wikipedia.org/wiki/High-level_programming_language), often [just-in-time compiled](https://en.wikipedia.org/wiki/Just-in-time_compilation), and [multi-paradigm](https://en.wikipedia.org/wiki/Programming_paradigm). It has [curly-bracket syntax](type), [dynamic typing](https://en.wikipedia.org/wiki/Dynamic_typing), prototype-based object-orientation, and [first-class functions](https://en.wikipedia.org/wiki/First-class_function).

Alongside HTML and [CSS](https://en.wikipedia.org/wiki/CSS), JavaScript is one of the core technologies of the [World Wide Web](https://en.wikipedia.org/wiki/World_Wide_Web). Over 97% of websites use it client-side for web page behavior, often incorporating third-party [libraries](https://en.wikipedia.org/wiki/Library_(computing)). Most web browsers have a dedicated JavaScript engine to execute the code on the [user](https://en.wikipedia.org/wiki/User_(computing))'s device.

## **2.4 PYTHON**

Python is a [high-level](https://en.wikipedia.org/wiki/High-level_programming_language), [general-purpose programming language](https://en.wikipedia.org/wiki/General-purpose_programming_language). Its design philosophy emphasizes [code readability](https://en.wikipedia.org/wiki/Code_readability) with the use of [significant indentation](https://en.wikipedia.org/wiki/Off-side_rule). Its [language constructs](https://en.wikipedia.org/wiki/Language_construct) and [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming) approach aim to help [programmers](https://en.wikipedia.org/wiki/Programmers) write clear, logical code for small- and large-scale projects.

Python is [dynamically-typed](system) and [garbage-collected](https://en.wikipedia.org/wiki/Garbage_collection_(computer_science)). It supports multiple [programming paradigms](https://en.wikipedia.org/wiki/Programming_paradigm), including s[tructured](https://en.wikipedia.org/wiki/Structured_programming) (particularly [procedural](https://en.wikipedia.org/wiki/Procedural_programming)), object-oriented and [functional programming](https://en.wikipedia.org/wiki/Functional_programming). It is often described as a "batteries included" language due to its comprehensive [standard library](https://en.wikipedia.org/wiki/Standard_library).

**Python Libraries**

a) Seaborn:

Seaborn is a Python data visualization library based on [matplotlib](https://matplotlib.org/). It provides a high-level interface for drawing attractive and informative statistical graphics. For a brief introduction to the ideas behind the library, you can read the [introductory notes](https://seaborn.pydata.org/introduction.html) or the [paper](https://joss.theoj.org/papers/10.21105/joss.03021). Visit the [installation page](https://seaborn.pydata.org/installing.html) to see how you can download the package and get started with it. You can browse the [example gallery](https://seaborn.pydata.org/examples/index.html) to see some of the things that you can do with seaborn, and then check out the [tutorial](https://seaborn.pydata.org/tutorial.html) or [API reference](https://seaborn.pydata.org/api.html) to find out how.

b) Matplotlib:

Matplotlib is a [plotting](https://en.wikipedia.org/wiki/Plotter) l[ibrary](https://en.wikipedia.org/wiki/Library_(computer_science)) for the [Python](https://en.wikipedia.org/wiki/Python_(programming_language)) programming language and its numerical mathematics extension [NumPy](https://en.wikipedia.org/wiki/NumPy). It provides an [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming) [API](https://en.wikipedia.org/wiki/API) for embedding plots into applications using general-purpose [GUI toolkits](https://en.wikipedia.org/wiki/GUI_toolkit) like [Tkinter](https://en.wikipedia.org/wiki/Tkinter), [wxPython](https://en.wikipedia.org/wiki/WxPython), [Qt](https://en.wikipedia.org/wiki/Qt_(software)), or [GTK](https://en.wikipedia.org/wiki/GTK). There is also a [procedural](https://en.wikipedia.org/wiki/Procedural_programming) "pylab" interface based on a [state machine](https://en.wikipedia.org/wiki/State_machine) (like [OpenGL](https://en.wikipedia.org/wiki/OpenGL)), designed to closely resemble that of [MATLAB](https://en.wikipedia.org/wiki/MATLAB), though its use is discouraged. [SciPy](https://en.wikipedia.org/wiki/SciPy) makes use of Matplotlib.

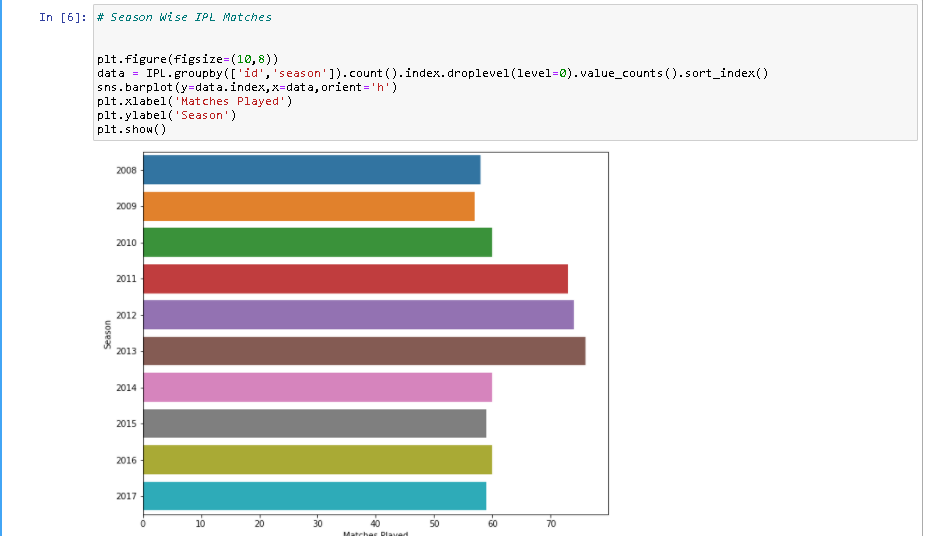


Fig. 1 Yearly Match Analysis using Matplotlib & Seaborn

c) Pandas:

Pandas is one of the powerful open source libraries in the Python programming language used for data analysis and data manipulation. If you want to work with any tabular data, such as data from a database or any other forms (Like CSV, JSON, Excel, etc.,) then pandas is the best tool.

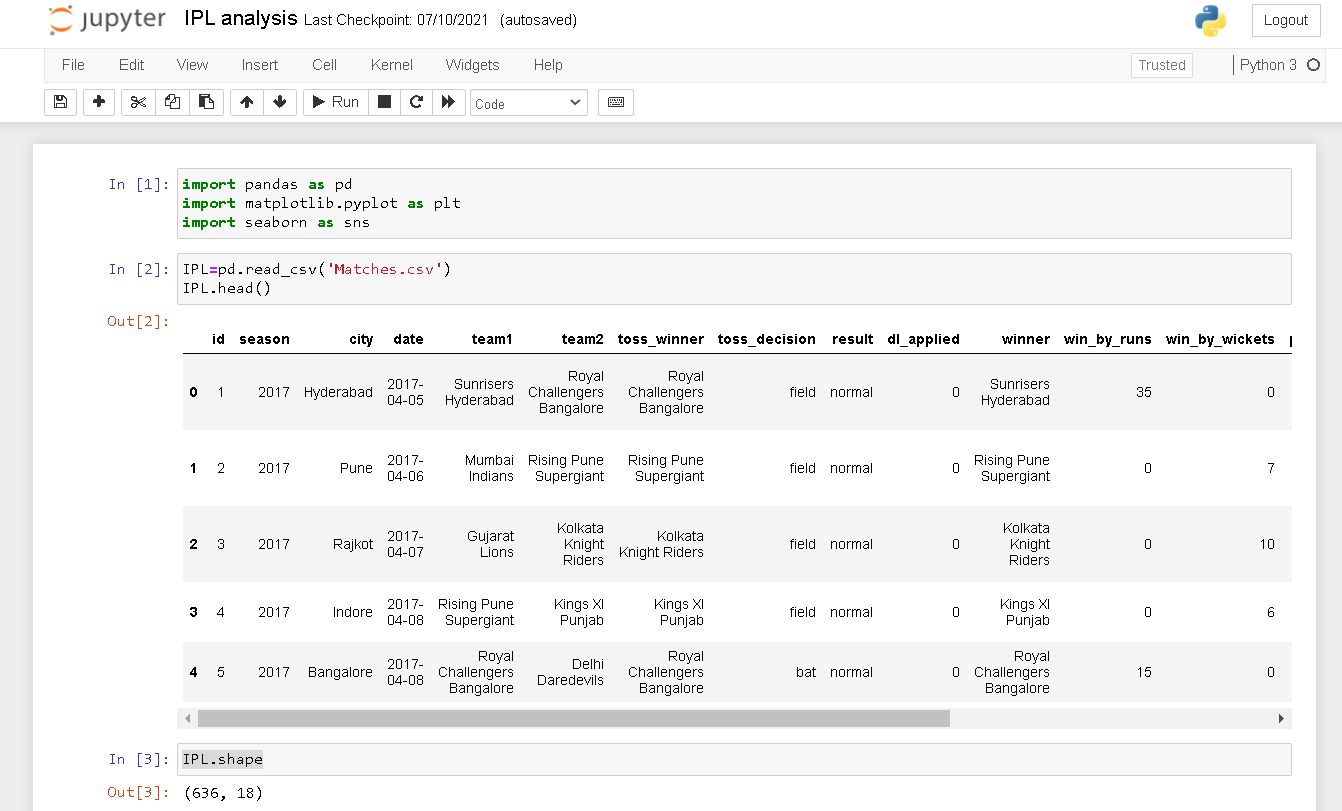
Pandas will reduce the complexity and make our work easy, and it can be applicable to any type of data that is ordered and unordered. The output of the pandas is also a tabular form named DataFrame. We can plot some Visualization graphs by using Matplotlib which is also a python library, it provides different plotting types such as scatter, bar, boxplot, etc.

Fig. 2 Creating Dataframe using Pandas

d) Flask:

Flask is a web framework, it’s a Python module that lets you develop web applications easily. It’s has a small and easy-to-extend core: it’s a microframework that doesn’t include an ORM (Object Relational Manager) or such features.

It does have many cool features like url routing, template engine. It is a WSGI web app framework.

**2.5 Jupyter Notebook:**

The Jupyter Notebook is an open source web application that you can use to create and share documents that contain live code, equations, visualizations, and text. Jupyter Notebook is maintained by the people at [Project Jupyter](http://jupyter.org/).

Jupyter Notebooks are a spin-off project from the IPython project, which used to have an IPython Notebook project itself. The name, Jupyter, comes from the core supported programming languages that it supports: Julia, Python, and R. Jupyter ships with the IPython kernel, which allows you to write your programs in Python, but there are currently over 100 other kernels that you can also use.

## **2.5 Deployment**

Deployment is the mechanism through which applications, modules, updates, and patches are delivered from developers to users. The methods used by developers to build, test and deploy new code will impact how fast a product can respond to changes in customer preferences or requirements and the quality of each change.

a) heroku:

[Heroku is a container-based cloud Platform as a Service (PaaS). Developers use Heroku to deploy, manage, and scale modern apps. Our platform is elegant, flexible, and easy to use, offering developers the simplest path to getting their apps to market.](https://en.wikipedia.org/wiki/Heroku" \l "cite_note-4)

Heroku is fully managed, giving developers the freedom to focus on their core product without the distraction of maintaining servers, hardware, or infrastructure. The Heroku experience provides services, tools, workflows, and polyglot support—all designed to enhance developer productivity.

* The deployment template will use the default PostgreSQL plan, which is the Hobby Dev (free) plan. You can upgrade to a larger plan afterwards as detailed in the [Herosku doc](https://devcenter.heroku.com/articles/updating-heroku-postgres-databases). Another option is to clone the wiki-heroku repository and edit the manifest to use the specific plan you need.
* While the initial deployment is extremely quick and easy, upgrading to a new version is not. You'll need to push a new commit to your app git repo in order to trigger a new build. Unless you are comfortable with the Heroku workflow for deployments, you may want to look into alternative hosting solutions instead.
* The free dyno plan (default) will "sleep" after 30 mins of inactivity. The first page load after such event will take several seconds to complete, as it must spin a new instance. Upgrade to a paid dyno plan to remove this limitation.
* The database endpoint must be provided as DATABASE\_URL environment variable to the app (default)

https://first-inning-score-predictions.herokuapp.com/

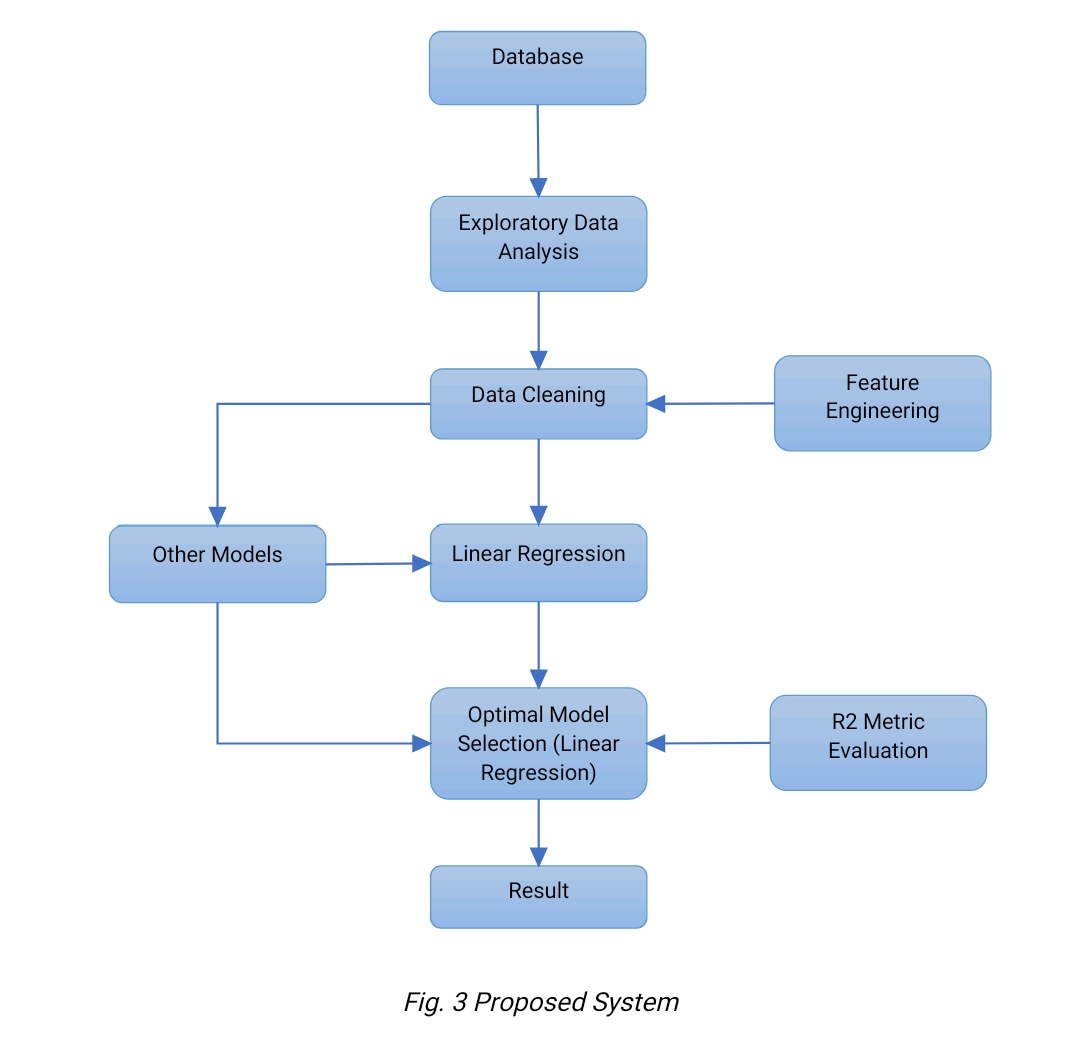
# **Chapter 3 :**

# **System Analysis**

## **3.1 PROPOSED SYSTEM**

The literature survey concluded that there was a need for a machine learning model which could predict the outcome of an IPL match before the game begins. Among all formats of cricket, Twenty20 format sees a lot of turnarounds in the momentum of the game. An over can completely change a game. Hence, predicting an outcome for a Twenty20 game is quite a challenging task. Besides, developing a prediction model for a league which is wholly based on auction is another hurdle. IPL matches cannot be predicted simply by making use of statistics over historical data solely. Because of players going under auctions, the players are bound to change their teams; which is why the ongoing performance of every player must be taken into consideration while developing a prediction model.

In sports, most of the prediction job is done using regression or classification tasks, both of which come under supervised learning. In simple terms, y = f(x) is a prediction model which is learned by the learning algorithm from a set of dataset: D = ((X1, y1),(X2, y2),(X3, y3), ...(Xn, yn)). Based on the type of output (y) supervised learning is divided further into two categories, viz., regression, and classification. In Regression, the output is a continuous value; however, classification deals with discrete kind of output. For predicting continuous values, Linear Regression appeared to be quite effective, and for classification problems like predicting the outcome of matches or classifying players.



### **3.1.1 ALGORITHMS & TECHNIQUES**

**A) Linear Regression:**

In linear regression, relationships are modeled using linear prediction functions whose unknown model parameters are estimated from the data. These models are called linear models. Like all forms of regression distribution of a response given the values of predictors, rather tham the common probability distribution of all these variables, which is the domain of multiverse analysis. The model has the form Y = a+ bX.

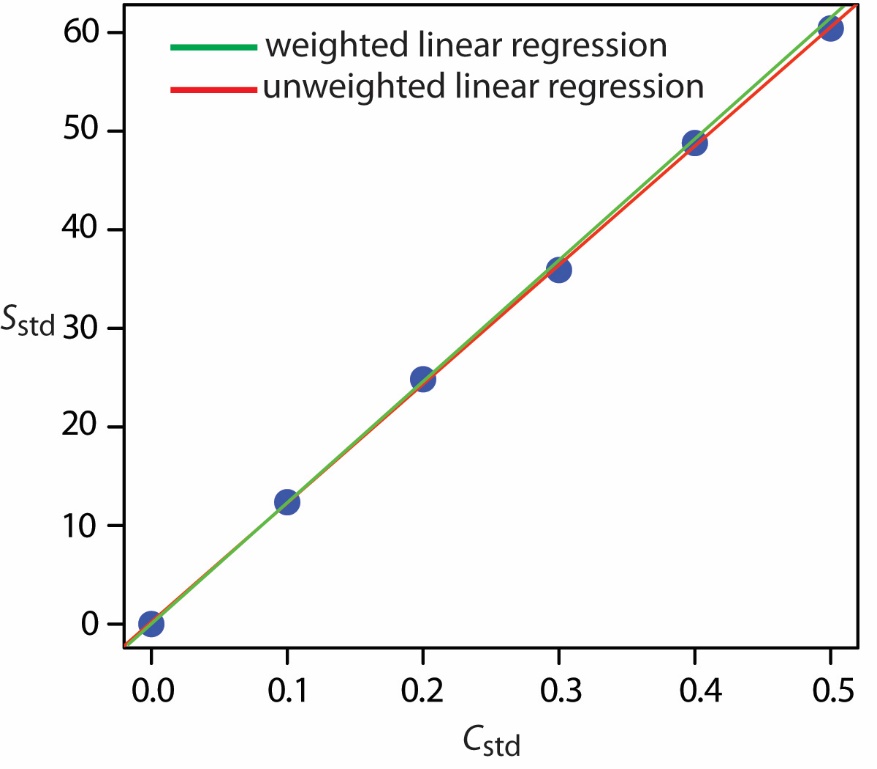


Fig. 3 Linear Regression

**B) Multilayer Perception Neural Network**

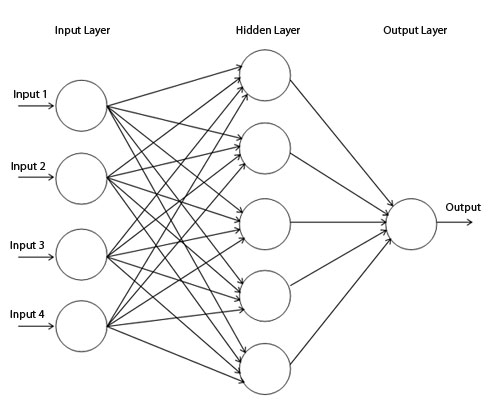
An MLP has minimum of three layers of nodes i.e an input layer, a hidden layer and an output layer. E ach node is a neuron that uses a nonlinear activation function except for the input nodes. MLP uses a supervised learning technique called backpropagation for training. Its multiple layers and non-linear activation distinguish MLP from a linear perceptron. It can help to distinguish the data that is not linearly separable.

Fig. 4 Multilayer Perception Neural Network

## **3.2 REQUIREMENT ANALYSIS**

### **3.2.1 HARDWARE REQUIREMENTS**

* Processor : Intel Core 2 Duo and onwards/ Ryzen
* RAM : 4 GB min
* HDD/SSD : 50 GB min
* Input Device : Mouse, Keyboard, Touchpad, Scanner
* Output Device: Monitor, Laptop

### **3.2.2 SOFTWARE REQUIREMENTS**

For the effective implementation of the new system, the following software has to be installed on the computer.

* Operating System : Windows XP, Windows Vista, Windows 7 and further versions, Linux.
* Web Server : Apache 2 HTTP Server
* Backend : Python
* Frontend : Microsoft Visual Studio Code 2019 for HTML, CSS & JS
* Web Browser : Microsoft Internet Explorer, Microsoft Edge, Mozilla, Google Chrome
* Deployment : Heroku, Github

## **3.3 PLANNING AND SCHEDULING**

Gantt chart is used for analyzing and planning complex projects. They help in planning each and every step of our task that is needed to be completed. It gives a schedule for task performance and completion. Gantt chart also helps us to see whether the project is on schedule. The main objective of the Gantt chart is to plan the task properly and to give proper time to each task.

**1. Gnatt Char**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| OBJECTIVE: IPL Analysis With Data Science | | | | | | | | | |
| Activity | Time Period | | | | | | | | |
| 07/21 | 08/21 | 09/21 | 10/21 | 11/21 | 12/21 | 01/22 | 02/22 | 03/22 |
| Project Proposal writing |  |  |  |  |  |  |  |  |  |
| Data Collection |  |  |  |  |  |  |  |  |  |
| Data Analysis & Requirement Specification |  |  |  |  |  |  |  |  |  |
| Conceptual Diagram |  |  |  |  |  |  |  |  |  |
| System design & Planning |  |  |  |  |  |  |  |  |  |
| Coding & Implementation |  |  |  |  |  |  |  |  |  |
| System Testing |  |  |  |  |  |  |  |  |  |
| System & Report presentation |  |  |  |  |  |  |  |  |  |

## **3.4 CONCEPTUAL MODEL**

### **3.4.1 Activity Diagram**

Activity diagram is an important diagram in UML. It represents the flow of each activity in the form of flow chart. The flow of activity is shown one below the other to understand in a better way. Activity diagram gives a quick look through our project's entire activity plan. Activity diagrams symbols meanings and rules:

**a) Start**

The start node represents the start of a particular activity, it is represented as a filled circle.

**b) End**

The end node represents the end of the activity, the entire system has only one end node. It is represented as a circle with a part of it as filled

**c) Activity**

This is an important component. It represents an activity

**d) Connector**

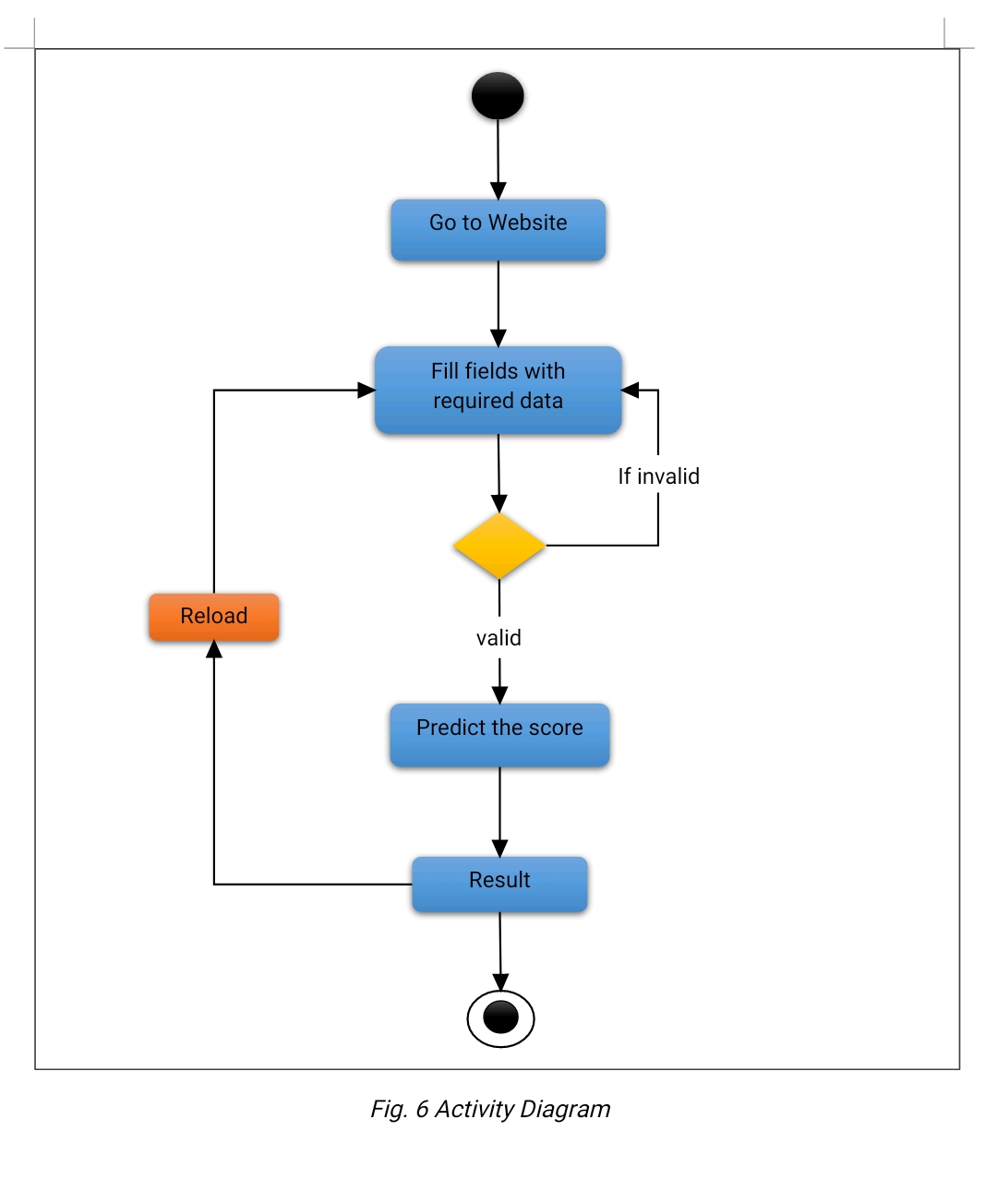
A connector in the activity diagram represents the flow of the system i.e. stepwise execution of various functions of the system

**e) Decision**

The decision part is represented by a diamond shape which acts as a decision-maker based on some conditions and do the actions accordingly.

**h) Flow final**

Flow final node is a node which terminates an activity flow. It destroys all the activity that arrives at it but has no influence on other flows in the activity. The notation is a circle with X inside.



### **3.4.2 Use Case Diagram**

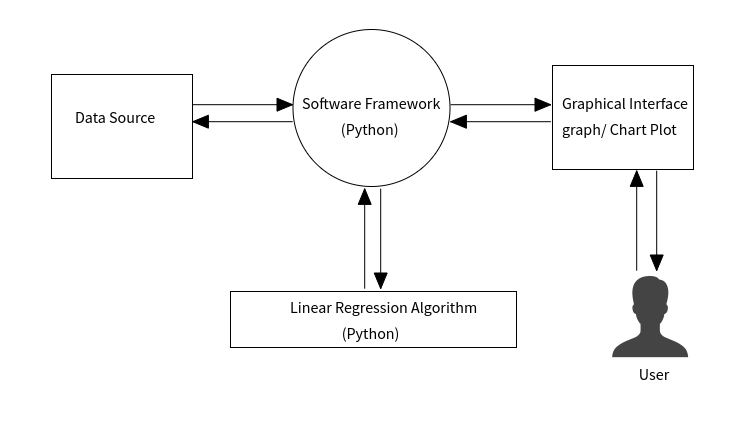


Fig. 5 Use Case Diagram

# **Chapter 4 :**

# **System Design**

## **4.1 MODULE DIVISION**

This project is developed using Visual Studio Code 2019. It uses Python as a backend. This chapter labels the preferred features and operations in aspect, comprising of screen layouts, business rules, process diagrams, pseudocode and other documentation. The application is built in a modular form where these functionalities are built into modules. The application includes the following modules:

* Home Page Module
* Result /Output Module

**1) Home Page Module**

In this module user is required to fill up the input text-fields with valid data like cricket team names, score, wickets, overs, etc. If the user enters invalid data then a prompt will be displayed on the top of home page. The user then must click on predict score after entering valid credentials inorder to get the output. The module name is “index.html”.

Fill-up the text fields

Click on predict score

**2) Result Module**

The result for the given input is displayed in this module. If the user wants to predict the score again then he/she have to click on the reload button given in the bottom of this module. The module name is ‘result.html”.

## **4.2 PROCEDURAL DESIGN**

### **4.2.1 DATA FLOW DIAGRAM**

A data flowchart (DFD) may be a graphical illustration of the "flow" of knowledge through a data system, modelling its method aspects. A DFD is commonly used as a primary step to make a summary of the system while not going into deep within the structure. DFDs may be used for the mental image of knowledge process (structured design). The on top of knowledge flowchart provides the data regarding the every activity is reticulated with one another and the way the flow of the information goes once user begin the appliance .When the user begin the appliance initial time user must login within the system then it verifies whether or not the user is registered within the system or not if user verified then next activity can get open. The additional elaborative description is within the abstract model section.

Rules of DFD diagram:

1. Maintain scope of the system suggests that of context diagram.

2. Maintain DFD so main consecutive sequence of the actions.

3. Scan from left to right and prime to bottom.

4. Acknowledge all inputs or outputs.

5. Acknowledge and Label from every method internal to the system together with conic section reasonably circle.

Data objects displayed by tagged arrows or transformation square measure outlined by circles known as bubble and DFD is given as hierarchy fashion that's the primary knowledge flow model outlined the system as an entire ensuing DFR routine context diagram provides increasing details with every consecutive level.

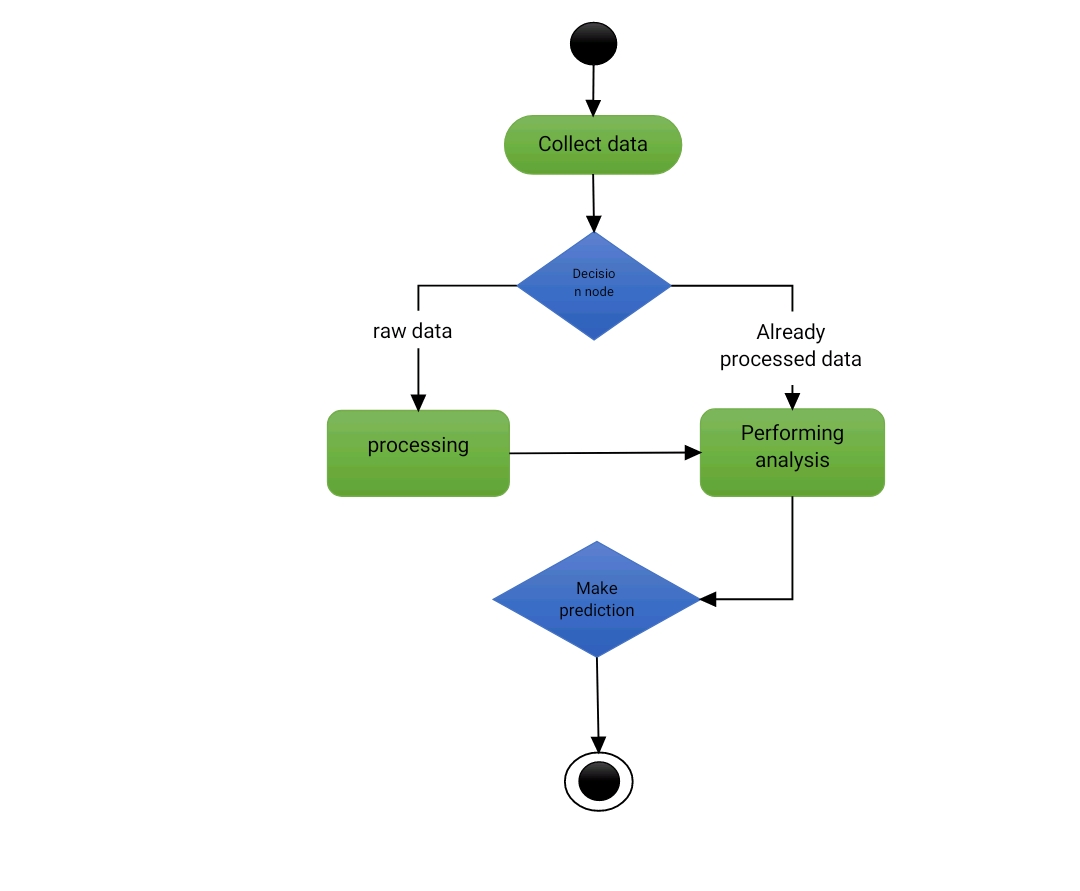
****

Fig 6 Data Flow Diagram

### **4.2.2 ALGORITHM DESIGN**

**Main Website Algorithm**

**STEP 1:** Go to the website.

**STEP 2:** Select the IPL Cricket to bowl & batting respectively.

**STEP 3:** Enter the number of overs(<5) and wickets taken in those.

**STEP 4:** Enter the current no. of runs scored.

**STEP 5:** Enter the runs scored and wickets taken in first 5 overs.

**STEP 6:** Finally click on predict score to get the output.

**Output/Result Algorithm**

**STEP 1:** Once the predicted score is displayed click on reload button to predict the score again.

**STEP 2:** Else close the tab in which you have opened the website.

## **4.3 USER INTERFACE DESIGN**

User Interface Layout is all about user’s interaction with the system. It does everything starting from logging into the system to getting the desired outputs. Following should be considered while designing

* User should know what he should do next.
* Default values to be entered by user for fields and answers should be specified.
* Messages or instructions should be kept short so that each user may read it.
* User must not be allowed to go further until they correct the error.
* User should not get any fatal errors.

The “UI” in UI design stands for “user interface.” The user interface is the graphical layout of an application. It consists of the buttons users click on, the text they read, the images, sliders, text entry fields, and all the rest of the item the user communicates with. This involves screen design, developments, interface animations, and all particular micro-interaction. Any sort of visual element, interaction, or animation must all be composed.

**1) Home Page**

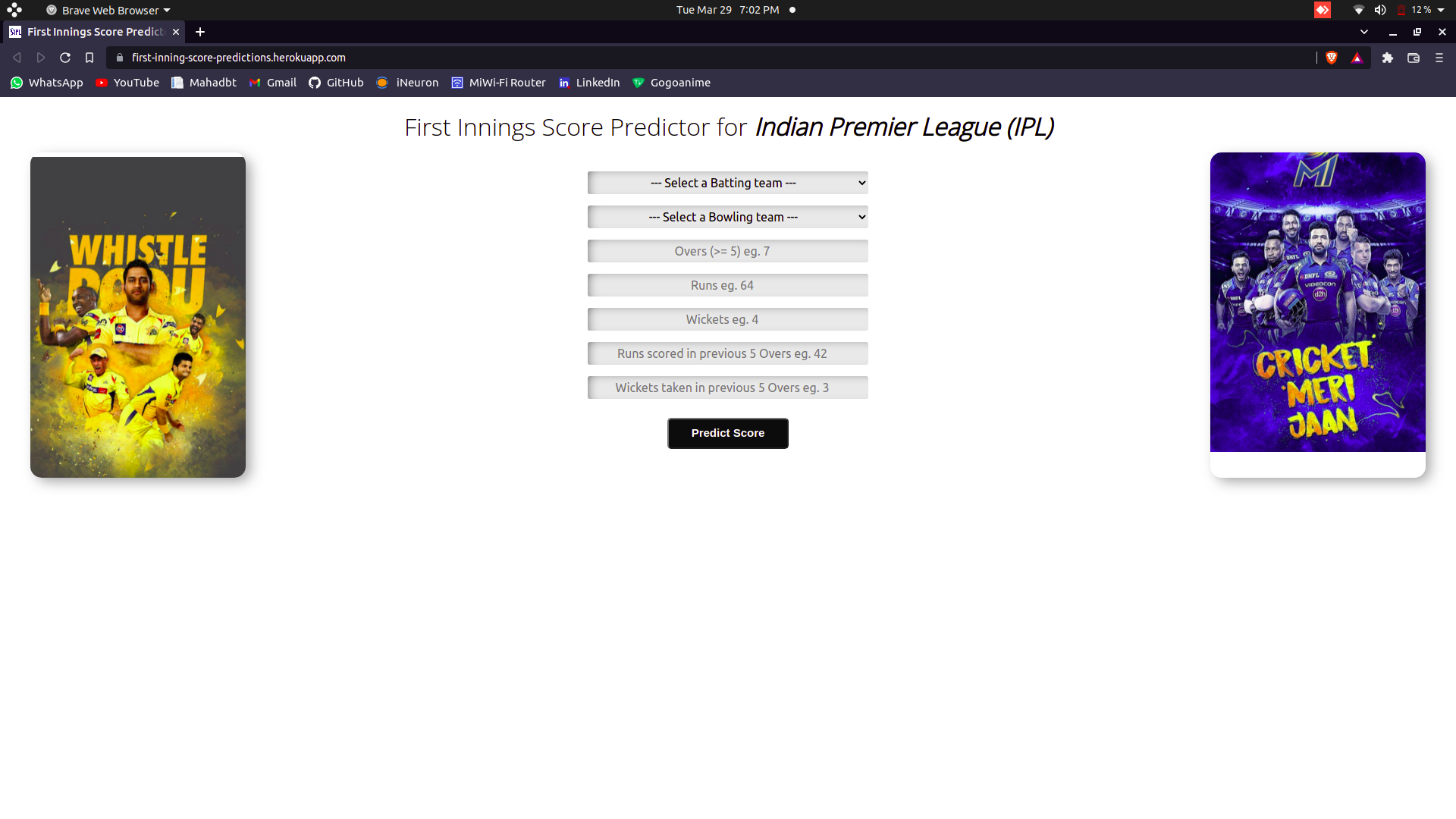
****

Fig. 7 Home Page

**2) Result Page**

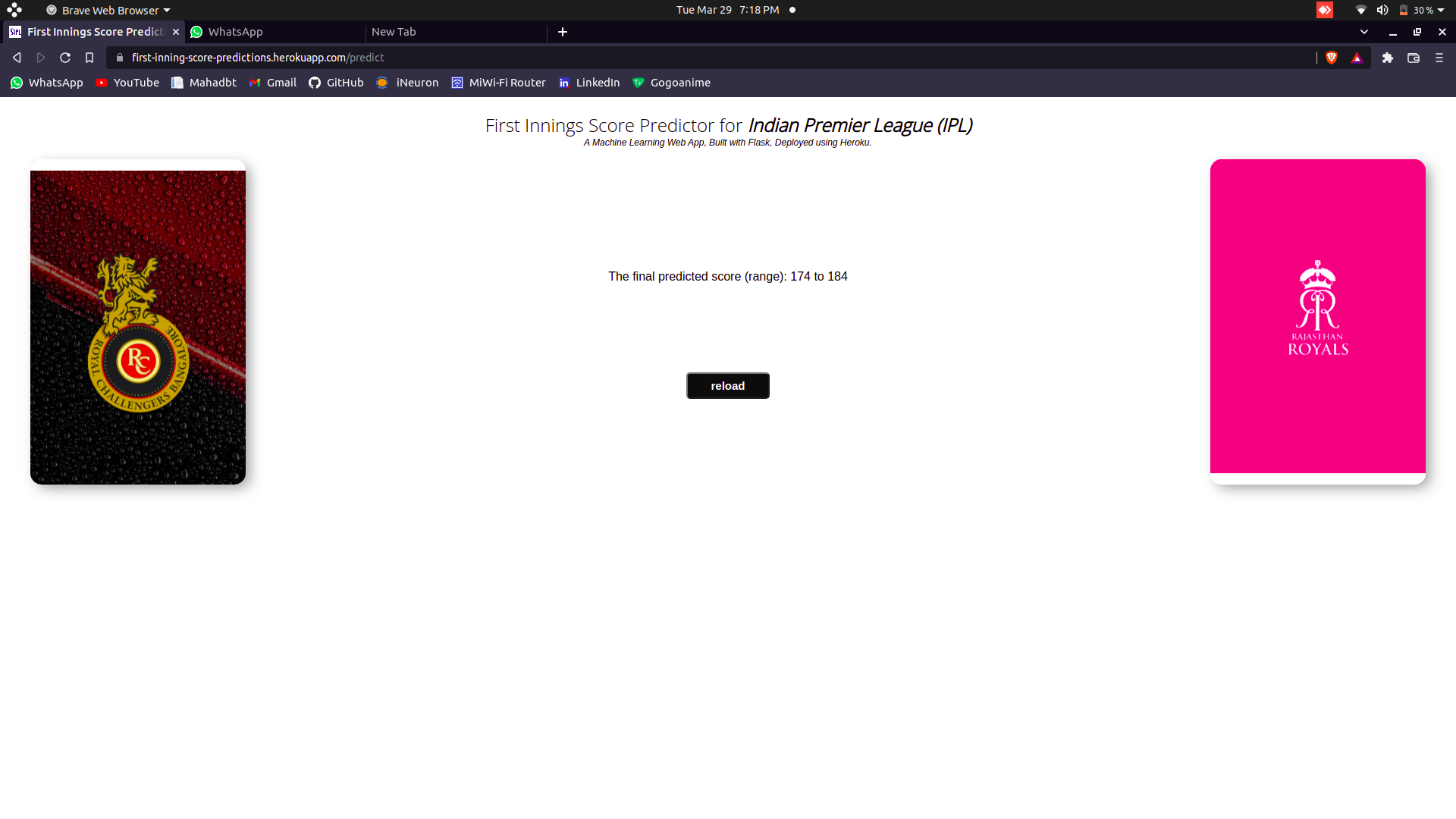
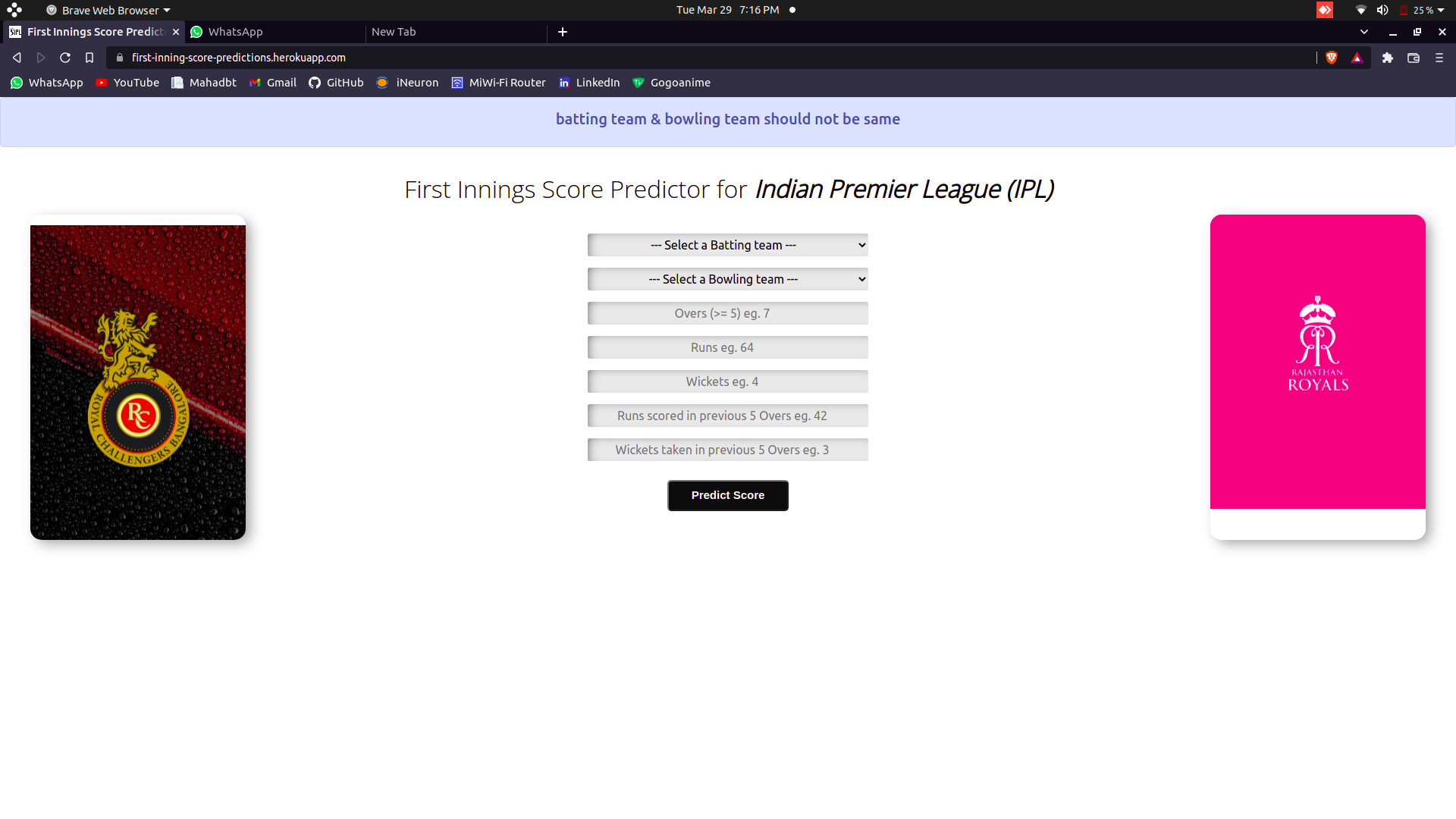


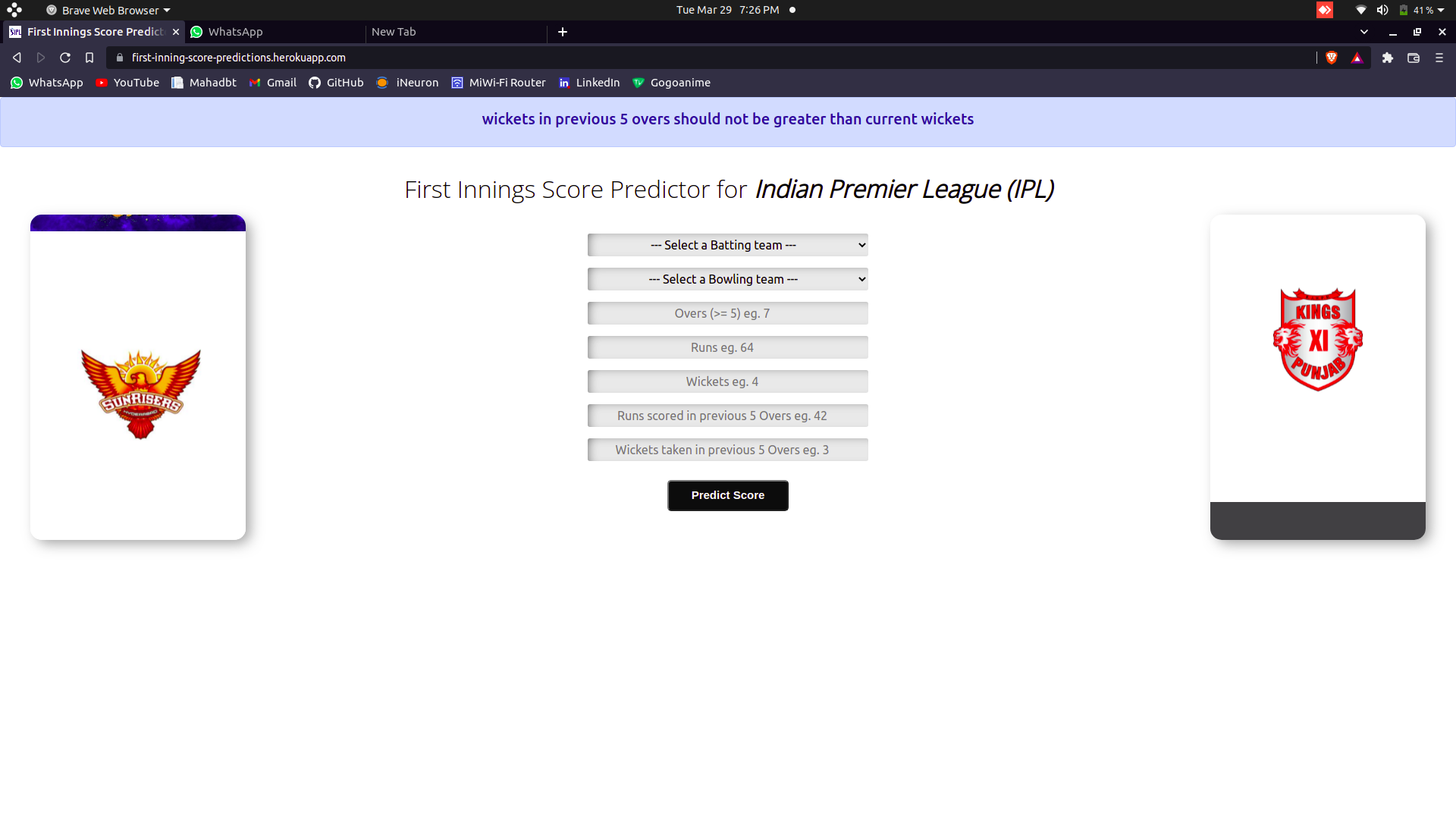
Fig. 8 Result Page

***3) Validation same team***

****

*fig .9 validation same team*

***4) Wickets validation***

*fig .10 wickets validation*

## **4.4 TEST CASE DESIGN**

Test case designing is done to solve test our program on assumptions of user input. Test cases are designed to help programmers solve errors they are a set of input or commands and the expected output cases are written down. If the project is passed in all the test cases then only it is sent to the user till then it is continuously modified and then again checked for errors and all the reports are written down. The various test cases required for this project and their set of inputs and outputs are given as follows:

2. **Test Case**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr No. | Name | Input | Expected Output | Actual Output | Remark |
| 1 | Home Page | Enter valid credentials | Navigate to result page | Navigate to result page | Pass |
| 2 | Home Page | Enter invalid credentials | Display error prompt | Display error prompt | Pass |
| 3 | Result Page | Click on predict button | Displays output | Displays output | Pass |
| 4 | Result page | Click on reload button | Navigate to home page | Navigate to home page | Pass |

# **Chapter 5 :**

# **Implementation and Testing**

## **5.1 IMPLEMENTATION APPROACHES**

The implementation is done via a web application which uses flask API, all the models are trained in a Jupyter Notebook and SKLearn library is used. Three models were built which gave us different score. The score metric used to evaluate all 3 models was R squared metric which is considered best for regression problems. As per the data, the linear regression shows the best fit to the data. Also the ridge regression is better for this prediction system compared to MLP Neural Networks. It can be seen that the R2 score for NN is 0.69 after convergence which is somewhere below 500 Epochs.

The best model i.e Linear Regression is then pickled and used in our web application to predict the model. Pickle is the library used for serialization and deserialization of python objects. When the user inputs the data into web application and submits it, following procedure occurs in python application which produces the output.

1) The batting team entered by user is assigned as 1 value in dataset and other batting teams are assigned as 0. Same thing happen with bowling teams.

2) A list is created with 16 variables eg. If Chennai is assigned as batting team and Mumbai is assigned as bowling team it will be hard coded in backend as [1,0,0,0,0,0,0,0] + [0,0,0,0,1,0,0,0] 3) This two lists are then joint and appended with remaining 5 variables input from HTML form. 4) The pickled model is then called by passing the above list (by converting into Numpy array) in it. The prediction of total score is then added by 5 to get maximum score possible and subtracted by 10 to get minimum score possible. This range is then displayed to user as final result. Eg. If the model predicts 150 as projected score for first inning then 140 to 155 will be the range of final result.

## **5.2 CODE DETAILS AND EFFICIENCY**

1) The following code displays the homepage user interface .

**index.html**

<div class='div2'>

<form action="{{ url\_for('predict') }}" method="POST">

<!-- Batting Team Dropdown -->

<select class="form-input align-center" name="batting-team">

<option value="none">--- Select a Batting team ---</option>

<option value="Mumbai Indians">Mumbai Indians</option>

<option value="Kolkata Knight Riders">Kolkata Knight Riders</option>

<option value="Chennai Super Kings">Chennai Super Kings</option>

<option value="Rajasthan Royals">Rajasthan Royals</option>

<option value="Kings XI Punjab">Kings XI Punjab</option>

<option value="Royal Challengers Bangalore">Royal Challengers Bangalore </option>

<option value="Delhi Daredevils">Delhi Daredevils</option>

<option value="Sunrisers Hyderabad">Sunrisers Hyderabad</option>

</select><br>

<!-- Bowling Team Dropdown -->

<select class="form-input align-center" name="bowling-team">

<option value="none">--- Select a Bowling team ---</option>

<option value="Mumbai Indians">Mumbai Indians</option>

<option value="Kolkata Knight Riders">Kolkata Knight Riders</option>

<option value="Chennai Super Kings">Chennai Super Kings</option>

<option value="Rajasthan Royals">Rajasthan Royals</option>

<option value="Kings XI Punjab">Kings XI Punjab</option>

<option value="Royal Challengers Bangalore">Royal Challengers Bangalore</option>

<option value="Delhi Daredevils">Delhi Daredevils</option>

<option value="Sunrisers Hyderabad">Sunrisers Hyderabad</option>

</select><br>

<input class="form-input" type="number" name="overs" placeholder="Overs(>=5) eg. 7" required><br>

<input class="form-input" type="number" name="runs" placeholder="Runs eg. 64" required><br>

<input class="form-input" type="number" name="wickets" placeholder="Wickets eg. 4" required><br>

<input class="form-input" type="number" name="runs\_in\_prev\_5"

placeholder="Runs scored in previous 5 Overs eg. 42" required><br>

<input class="form-input" type="number" name="wickets\_in\_prev\_5"

placeholder="Wickets taken in previous 5 Overs eg. 3" required><br>

<input type="submit" class="myButton" value="Predict Score">

</form>

</div>

2) The following code displays the javascript code .

**script.js**

$(function() {

setTimeout(function() { $("#alert").fadeOut(1500); }, 5000)

})

3) The following code displays the backend part written in python.

**app.py**

from flask import Flask, render\_template, request ,flash ,redirect, url\_for

import numpy as np

import pickle

filename = 'first-innings-score-lr-model.pkl'

regressor = pickle.load(open(filename, 'rb'))

app = Flask(\_\_name\_\_)

app.config['SECRET\_KEY'] = 'SECRET\_KEY'

@app.route('/')

def home():

return render\_template('index.html')

@app.route('/predict', methods=['POST'])

def predict():

temp\_array = list()

if request.method == 'POST':

batting\_team = request.form['batting-team']

if batting\_team == 'Chennai Super Kings':

temp\_array = temp\_array + [1,0,0,0,0,0,0,0]

elif batting\_team == 'Delhi Daredevils':

temp\_array = temp\_array + [0,1,0,0,0,0,0,0]

elif batting\_team == 'Kings XI Punjab':

temp\_array = temp\_array + [0,0,1,0,0,0,0,0]

elif batting\_team == 'Kolkata Knight Riders':

temp\_array = temp\_array + [0,0,0,1,0,0,0,0]

elif batting\_team == 'Mumbai Indians':

temp\_array = temp\_array + [0,0,0,0,1,0,0,0]

elif batting\_team == 'Rajasthan Royals':

temp\_array = temp\_array + [0,0,0,0,0,1,0,0]

elif batting\_team == 'Royal Challengers Bangalore':

temp\_array = temp\_array + [0,0,0,0,0,0,1,0]

elif batting\_team == 'Sunrisers Hyderabad':

temp\_array = temp\_array + [0,0,0,0,0,0,0,1]

bowling\_team = request.form['bowling-team']

if bowling\_team == 'Chennai Super Kings':

temp\_array = temp\_array + [1,0,0,0,0,0,0,0]

elif bowling\_team == 'Delhi Daredevils':

temp\_array = temp\_array + [0,1,0,0,0,0,0,0]

elif bowling\_team == 'Kings XI Punjab':

temp\_array = temp\_array + [0,0,1,0,0,0,0,0]

elif bowling\_team == 'Kolkata Knight Riders':

temp\_array = temp\_array + [0,0,0,1,0,0,0,0]

elif bowling\_team == 'Mumbai Indians':

temp\_array = temp\_array + [0,0,0,0,1,0,0,0]

elif bowling\_team == 'Rajasthan Royals':

temp\_array = temp\_array + [0,0,0,0,0,1,0,0]

elif bowling\_team == 'Royal Challengers Bangalore':

temp\_array = temp\_array + [0,0,0,0,0,0,1,0]

elif bowling\_team == 'Sunrisers Hyderabad':

temp\_array = temp\_array + [0,0,0,0,0,0,0,1]

overs = float(request.form['overs'])

runs = int(request.form['runs'])

wickets = int(request.form['wickets'])

runs\_in\_prev\_5 = int(request.form['runs\_in\_prev\_5'])

wickets\_in\_prev\_5 = int(request.form['wickets\_in\_prev\_5'])

## **5.3 TESTING APPROACH**

* **Beta Testing**

It is one of the Acceptance Testing types, which adds value to the product as the end-user (intended real user) validates the product for functionality, usability, reliability, and compatibility. It makes it less demanding to test the application on a large number of test machines and fix the issues previously discharging the application to the general population.

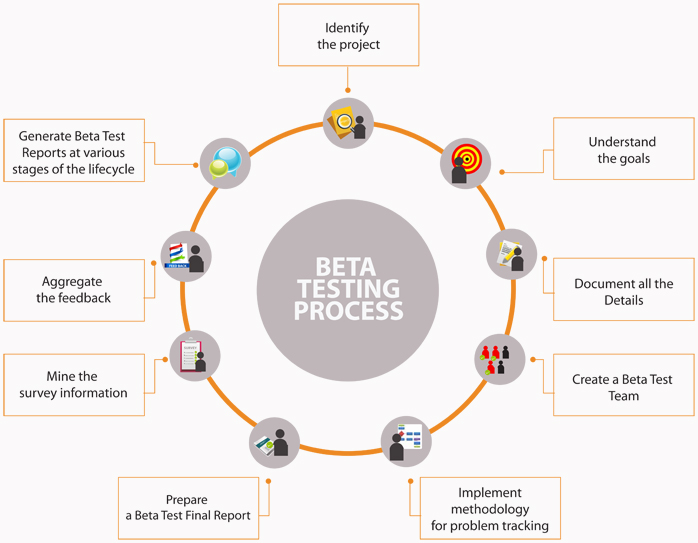


Fig. 9 Beta Testing

3 **Beta Testing**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr No. | User Acceptance | Active Value | Expected Value | Defect |
| 1 | Take valid data | No prompt displayed | No prompt displayed | 0 |
| 2 | Take invalid data | Prompt displayed successfully | Prompt displayed successfully | 0 |
| 3 | Project runs successfully | Home page of the project displayed | Home page of the project displayed | 0 |
| 4 | Predict the score | Score displayed successfully | Score displayed successfully | 0 |

## **5.4 MODIFICATIONS AND IMPROVEMENTS**

This is the basic and first version for this project. Modifications can be made and implemented in further advanced versions of this project. Some modifications that can be made are noted below:

1. The interface of this website can be made more user friendly.
2. Currently this website can predict score only for IPL matches. In future this system can be made to predict the scores of ODI matches, test matches.
3. It predicts score only for the 1st inning provided the information from first inning and 2nd inning provided with information from second inning.
4. This website can be modified to mobile application so that every user can be benefitted from this system.
5. We can add a login and register modules so that user can keep track of his/her activities.

# **Chapter 6 :**

# **Conclusion**

The proposed work can help us predict accurate projected score in between a progressing match. It is successfully implemented as a web application with the help of Flask. This model provides us with score as accurate as 75.226% with the help of linear regression. The above claim is verified by testing the model over the matches played between years 2017 to 2019. Following are the results of each model used for prediction.

This analysis can be done by considering more factors like weather, venue, batsman and bowler hence giving us better results. Also the previous match outcomes resulted due to super over must be separately analyzed and included in the model. This dynamic factors can change the outcome of the match in split seconds. Additional work can lead to obtain usa better model with much accurate predictions.

The Twenty20 format of cricket carries a lot of randomness, because a single over can completely change the ongoing pace of the game. Indian Premier League is still at infantry stage, it is just a decade old league and has way less number of matches compared to test and one-day international formats. Hence, designing a machine learning model for predicting the match outcome of an auction-based Twenty20 format premier league with an accuracy of 72.66% and F1 score of 0.72 is highly satisfactory at this stage.

# **Chapter 7:**

# **Reference**

## **7.1 BIBLIOGRAPHY**

* Parag Shah, “Predicting Outcome of Live Cricket Match Using Duckworth-Lewis Par Score”, Publisher: International Journal of Latest Technology in Engineering, Management & Applied Science, Volume VI, Issue VIIS, July 2017.
* Haseeb Ahmad, Ali Daud, Licheng Wang, Haibo Hong, Husain Dawood, and Yixian Yang , “Prediction of Rising Stars in the Game of Cricket”, Publisher: IEEE Access, Issue March 4 2017.
* Mehvish Khan, Riddhi Shah, “Role of External Factors on Outcome of One Day International Cricket (ODI) Match and Predictive Analysis”, Publisher: International Journal of Advanced Research in Computer and Communication Engineering, Vol. 4, Issue 6, June 2015.
* Akhil Nimmagadda et. Al, “Cricket score and winning prediction using data mining”, IJARnD Vol.3, Issue3.
* Rameshwari Lokhande and P.M.Chawan, “Live Cricket Score and Winning Prediction” International Journal of Trend in Research and Development, Volume 5(1), ISSN: 2394-9333.
* Amit Dhurandhar and Alin Dobra, “ Probabilistic Characterization of Random Decision Trees”, Journal of Machine Learning Research, 2008.

* <https://www.w3schools.com/>
* <https://www.tutorialspoint.com/>
* <https://docs.microsoft.com/en-in/>
* <https://www.youtube.com/>
* <https://www.google.com/>