

**A PROJECT REPORT
ON
PREDICTING BEST MATCH SPORTSPERSON FOR PRODUCT
ADVERTISEMENT**

**Submitted to
UNIVERSITY OF MUMBAI**

In Partial Fulfilment of the Requirement for the Award of

**BACHELOR'S DEGREE IN
COMPUTER ENGINEERING**

BY

Shah Shahil Shakir Hussain Hafizunnisa	14CO45
Sarguroh Junaid Jawed Sadaf	14CO43
Gavandi Abhay Audumbar Varsha	14CO22
Syed Areeb Iqbal Ahmad Rehana	13CO61

**UNDER THE GUIDANCE OF
Prof. Apeksha Gopale**



**DEPARTMENT OF COMPUTER ENGINEERING
Anjuman-I-Islam's Kalsekar Technical Campus
SCHOOL OF ENGINEERING & TECHNOLOGY**

**Plot No. 2 3, Sector - 16, Near Thana Naka,
Khandagaon, New Panvel - 410206**

2018-2019

**AFFILIATED TO
UNIVERSITY OF MUMBAI**

**A PROJECT II REPORT
ON**

**“PREDICTING BEST MATCH SPORTSPERSON FOR PRODUCT
ADVERTISEMENT”**

**Submitted to
UNIVERSITY OF MUMBAI**

In Partial Fulfilment of the Requirement for the Award of

**BACHELOR’S DEGREE IN
COMPUTER ENGINEERING**

BY

Shah Shahil Shakir Hussain Hafizunnisa	14CO45
Sarguroh Junaid Jawed Sadaf	14CO43
Gavandi Abhay Audumbar Varsha	14CO22
Syed Areeb Iqbal Ahmad Rehana	13CO61

**UNDER THE GUIDANCE OF
Prof. Apeksha Gopale**



**DEPARTMENT OF COMPUTER ENGINEERING
Anjuman-I-Islam's Kalsekar Technical Campus
SCHOOL OF ENGINEERING & TECHNOLOGY
Plot No. 2 3, Sector - 16, Near Thana Naka,
Khandagaon, New Panvel - 410206**

**2018-2019
AFFILIATED TO**



UNIVERSITY OF MUMBAI

Anjuman-i-Islam's Kalsekar Technical Campus

Department of Computer Engineering

SCHOOL OF ENGINEERING & TECHNOLOGY

Plot No. 2 3, Sector - 16, Near Thana Naka,

Khandagaon, New Panvel - 410206



CERTIFICATE

This is certify that the project entitled

‘Predicting Best Match Sportsperson for Product Advertisement’

submitted by

Shah Shahil Shakir Hussain Hafizunnisa	14CO45
Sarguroh Junaid Jawed Sadaf	14CO43
Gavandi Abhay Audumbar Varsha	14CO22
Syed Areeb Iqbal Ahmad Rehana	13CO61

is a record of bonafide work carried out by them, in the partial fulfilment of the requirement for the award of Degree of Bachelor of Engineering (Computer Engineering) at *Anjuman-I-Islam's Kalsekar Technical Campus, Navi Mumbai* under the University of MUMBAI. This work is done during year 2018-2019, under our guidance.

Date: / /

Prof. Apeksha Gopale
Project Supervisor

Prof. Kalpana Bodke
Project Coordinator

Prof. Tabrez Khan
HOD, Computer Department

DR. ABDUL RAZAK HONNUTAGI
Director

External Examiner

Acknowledgements

I would like to take the opportunity to express my sincere thanks to my guide **Prof. Apeksha Gopale**, Assistant Professor, Department of Computer Engineering, AIKTC, School of Engineering, Panvel for his invaluable support and guidance throughout my project research work. Without his kind guidance & support this was not possible.

I am grateful to him/her for his timely feedback which helped me track and schedule the process effectively. His/her time, ideas and encouragement that he gave is help me to complete my project efficiently.

We would like to express deepest appreciation towards **DR. ABDUL RAZAK HONNUTAGI**, Director, AIKTC, Navi Mumbai, **Prof. Tabrez Khan**, Head of Department of Computer Engineering and **Prof. Kalpana Bodke**, Project Coordinator whose invaluable guidance supported us in completing this project.

At last we must express our sincere heartfelt gratitude to all the staff members of Computer Engineering Department who helped me directly or indirectly during this course of work.

Shah Shahil Shakir Hussain Hafizunnisa 14CO45

Sarguroh Junaid Jawed Sadaf 14CO43

Gavandi Abhay Audumbar Varsha 14CO22

Syed Areeb Iqbal Ahmad Rehana 13CO61

Project I Approval for Bachelor of Engineering

This project entitled *Predicting Best Match Sportsperson for Product Advertisement* by *Shah Shahil Shakir Hussain Hafizunnisa(14CO45), Sarguroh Junaid Jawed Sadaf(14CO43), Gavandi Abhay Audumbar Varsha(14CO22), Syed Areeb Iqbal Ahmad Rehana(13CO61)* is approved for the degree of *Bachelor of Engineering in Department of Computer Engineering*.

Examiners

1.
2.

Supervisors

1.
2.

Chairman

.....

Declaration

I declare that this written submission represents my ideas in my own words and where others ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Shah Shahil Shakir Hussain Hafizunnisa
14CO45

Sarguroh Junaid Jawed Sadaf
14CO43

Gavandi Abhay Audumbar Varsha
14CO22

Syed Areeb Iqbal Ahmad Rehana
13CO61

B.E. (Computer Engineering)
University of Mumbai.

ABSTRACT

Sports are one of the popular forms of entertainment in today's world. People do like to express their views on social sites regarding sports, players etc. As we all know that people do watch television, advertisements and show interest in the products endorsed by their favourite sports person. The proposed system is considering the performance or ranking of a sports person and their popularity on social site to decide on the best suitable candidate for particular product endorsement in order to increase the sale of the product.

Keywords: Sentimental analysis, Machine Learning, Product Advertisement, Sports, Naive bayes, Prediction.

Contents

Acknowledgement	iii
Project I Approval for Bachelor of Engineering	iv
Declaration	v
Abstract	vi
Table of Contents	ix
1 Introduction	2
1.1 Purpose	2
1.2 Project Scope	2
1.3 Project Goals and Objectives	3
1.3.1 Goals	3
1.3.2 Objectives	3
1.4 Organization of Report	3
2 Literature Survey	5
2.1 Quantifying the Performance of Players in Football Match	5
2.1.1 Advantages of Paper	5
2.1.2 Disadvantages of Paper	5
2.1.3 How to Overcome the Problems Mentioned in Paper	6
2.2 Evaluating Player Performance via Statistical Network Modeling	6
2.2.1 Advantages of Paper	6
2.2.2 Disadvantages of Paper	6
2.2.3 How to Overcome the Problems Mentioned in Paper	6
2.3 Twitter Sentiment Analysis Using Hybrid Cuckoo Search Method	7
2.3.1 Advantages of Paper	7
2.3.2 Disadvantages of Paper	7
2.3.3 How to Overcome the Problems Mentioned in Paper	7
2.4 Technical Analysis Of Player's Performance	8
2.4.1 Advantages of Paper	8
2.4.2 Disadvantages of Paper	8
2.4.3 How to Overcome the Problems Mentioned in Paper	8
2.5 Technical Review	8
2.5.1 Advantages of Technology	9
2.5.2 Reasons to Use This Technology	9

3	Project Planning	10
3.1	Members and Capabilities	10
3.2	Roles and Responsibilities	10
3.3	Assumptions and Constraints	10
3.3.1	Assumptions	10
3.3.2	Constraints	10
3.4	Project Management Approach	11
3.4.1	Planning	11
3.4.2	Risk Analysis	12
3.4.3	Engineering Phase	12
3.4.4	Evaluation	12
3.5	Ground Rules for the Project	12
3.6	Project Budget	12
3.7	Project Timeline	13
4	Software Requirements Specification	14
4.1	Overall Description	14
4.1.1	Product Perspective	14
4.1.2	Product Features	14
4.1.3	User Classes and Characteristics	15
4.1.4	Operating Environment	15
4.1.5	Design and Implementation Constraints	15
4.2	System Features	15
4.2.1	System Feature	15
4.3	External Interface Requirements	17
4.3.1	User Interface	17
4.3.2	Hardware Interface	17
4.3.3	Software Interface	17
4.3.4	Communications Interfaces	17
4.4	Nonfunctional Requirements	17
4.4.1	Performance Requirements	17
4.4.2	Safety Requirements	17
4.4.3	Security Requirements	17
5	System Design	19
5.1	System Requirements Definition	19
5.1.1	Functional Requirements	19
5.1.2	System Requirements (non-functional requirements)	23
5.2	System Architecture Design	24
5.3	Sub-system Development	25
5.3.1	Fetching Data	25
5.3.2	Dataset	26

5.3.3	Filtering of Data	27
5.4	Systems Integration	28
5.4.1	Class Diagram	29
5.4.2	Sequence Diagram	30
5.4.3	Component Diagram	31
5.4.4	Deployment Diagram	32
6	Implementation	33
6.1	Scrapping From Cricbuzz	33
6.2	Fetching Data From Tweepy	35
6.3	Insertion of Cricbuzz Data to Database(Python Script)	37
6.4	Cricbuzz Different Data Merge Into One Super Data	39
6.5	Removing Tuple Duplication- Final Rank (Averaging all Data) . . .	41
6.6	Final Rank Player Details to Tweets Table	42
6.7	Updating Positive Sentiment and Overall Sentiment Type Into Tweets Table by Using Tweepy Script	43
7	System Testing	46
7.1	Test Cases and Test Results	46
7.2	Test Case	46
8	Screenshots of Project	50
8.1	Front End of the System	50
8.2	Database Storage	55
8.3	Trending Player and Popular Facelist	56
9	Conclusion and Future Scope	57
9.1	Conclusion	57
9.2	Future Scope	57
	References	58
	Achievements	58

List of Figures

3.1	Model of our project	11
5.1	Use Case	20
5.2	DFD Level 0	21
5.3	DFD Level 1	22
5.4	DFD Level 2	23
5.5	System Architecture	24
5.6	Fetching data	26
5.7	Dataset	27
5.8	Filtering of data	28
5.9	Class Diagram	29
5.10	Sequence diagram stage 1	30
5.11	Sequence diagram stage 2	30
5.12	Component Diagram	31
5.13	Deployment Diagram	32
6.1	Scraping script from Cricbuzz	33
6.2	Fetches data from twitter	35
6.3	Sentimental analysis pie-chart	35
6.4	Insertion of cricbuzz data in database	37
6.5	All-in-One Data For Selection	39
6.6	Final Rank Table	41
6.7	Combining rank table and tweets table	42
6.8	Updation of table	43
7.1	Successful loading of website	48
7.2	Successful scraping of the data	48
7.3	Successful data storage	49
8.1	Home Page	50
8.2	Login Page	51
8.3	Registration Page	51
8.4	Paid user home page	52
8.5	Profile page	52
8.6	Payment gateway page	53

8.7	Payment gateway authenticator page	53
8.8	Subscription page	54
8.9	Pay us page	54
8.10	Pay us confirmation	55
8.11	Stored database	55
8.12	Trending players facelist	56
8.13	Vendors filtered popular facelist	56

List of Tables

3.1	Table of Capabilities	10
3.2	Table of Responsibilities	10
7.1	Table for Test Cases and Results	46

Chapter 1

Introduction

Our project is based on 3 key factors : Celebrities, Social Media, Advertising. We all have an idol in our life whom we admire, or like to become like, and as a result we try to imitate them or watch whatever they do on television or other platforms. Social media is the oxygen to youngsters nowadays. We like to give our opinions on everything happening in our lives, updating statuses and commenting on whatever we relate to. The last part is Advertising. Advertising is the norm of the business industries nowadays. A product cannot sell without a proper advertisement. Now if we combine all the three factors, we are trying to build a web application that predicts the best match sportsperson for a product through analysis of all the three key factors.

1.1 Purpose

The purpose of this project is to foresee or predict popular sports person for a product and to give a head start to the vendor for grabbing a player before the rival competition. The other purpose for this project is to be a start of a revolution in the advertising market and change the way how agencies work..

1.2 Project Scope

This project will consist of creating a marketable software for advertising agencies based upon the popularity of a sportsperson. The project will be completed by 30th March 2019. Modules of the project will include extraction of tweets, conversion of tweets into meaningful datasets and taking agencies requirements into consideration for generating best sportsperson for product advertisement.

1.3 Project Goals and Objectives

1.3.1 Goals

1. **Best sportsperson for the product:-**Through our software we are going to determine suitable player for the advertising agency according to their requirements, budget and scale in profit.
2. **Rise in product sales:-** Selecting a suitable sportsperson who is trending can lead the companies or advertising agency with a boost in their product sales which will lead to a successful campaign.
3. **Profit for both:-** This leads to the profit of both the sportsperson as well as the advertising agency, as the sportsperson gains prominence as well as money and for the advertising agency it leads to a profitable quarter.

1.3.2 Objectives

Through our product, we will decide appropriate player for the publicizing vendor as per their prerequisites, spending plan and scale in benefit.

Choosing an appropriate sportsperson who is inclining can lead the organizations or publicizing vendor with a lift in their item deals which will prompt a fruitful battle.

This prompts the benefit of both the sportsperson just as the publicizing organization, as the sportsperson gains conspicuousness just as cash and for the promoting vendor it prompts a beneficial quarter.

1.4 Organization of Report

Chapter 1 gives a brief introduction about our project.

Chapter 2 describes the literature review of the papers that existed for our references to build our project.

Chapter 3 talks about the project planning and different roles and capabilities of the team member.

Chapter 4 describe the brief description of the srs and the other requirement of the project.

Chapter 5 shows the system design, functional requirement and different diagram of the projects.

Chapter 6 shows the implementation of the different programs of our project.

Chapter 7 shows the different testings performed and the problems faced. It also shows the snapshots of the current working application.

Chapter 8 is the closure to the book and tries to conclude the work in the project and also mentions the future scope as to where it would be used.

Chapter 9 is a step by step guide about using the final product.

Chapter 2

Literature Survey

2.1 Quantifying the Performance of Players in Football Match

It's hard to evaluate the impact of a player's performance on the team. The tradition method and rating systems involve looking at a few metrics which include goal scored, assists, key passes, tackles, intercept, etc. Often this methodology makes the goal scorer and the assist provider the most important players of the team, which might not always be the case. No wonder Ballon d'Or winners are forwards and not defenders. These numbers make sense when comparing similar metrics. However, when comparing a forward, whose primary job is to score and assist goals, with a defender, whose primary task is to clear the ball, tackle, it's difficult[1]. Football is a team sport; there is a complex interaction between the players. A winning goal might be a result of a threading pass by the midfielder and the winger making a diagonal run to take one of the central defenders with him creating space for the striker to score the goal. In such complicated scenario, it is tough to allocate the contribution of the goal. The paper discussed a statistical method, using regression and optimization, to qualitatively allocate the points contributed to the team by a particular player during a season. Thus, even thou the player who scored the winning goal might have secured 3 points for his team; his contribution to the team is not 3 points. The paper provides a methodology for distributing those 3 points to their rightful contributors[1].

2.1.1 Advantages of Paper

Predicting the impact of player: The software is having the capability to use divination to predict the impact of a player in the upcoming football match.

2.1.2 Disadvantages of Paper

Inaccurate results: The main problem is that it can sometimes give inaccurate results due to the large sum of datasets present as the mathematical formula can vary due to the huge amount of data.

2.1.3 How to Overcome the Problems Mentioned in Paper

Naive Bayes Algorithm: Naive Bayes Algorithm is a family of simple probabilistic classifiers which can accept large datasets and provide accurate results in return.

2.2 Evaluating Player Performance via Statistical Network Modeling

The major difficulty in evaluating individual player performance in basketball is adjusting for interaction effects by teammates. With the advent of play-by-play data, the plus-minus statistic was created to address this issue. While variations on this statistic do correct for some existing confounders, they struggle to gauge two aspects: the importance of a player's contribution to his units or squads, and whether that contribution came as unexpected (i.e. over or under-performed) as denoted by a statistical model[2]. We quantify both in this paper by adapting a network-based algorithm to estimate centrality scores and their corresponding statistical significances. Using four seasons of data, we construct a single network where the nodes are players and an edge exists between two players if they played in the same team unit. These edges are assigned weights that correspond to an aggregate sum of the two players' performance during the time they played together. We determine the statistical contribution of a player in this network by the frequency with which that player is visited in a random walk on the network, and we implement bootstrap techniques on these original weights to produce reference distributions for testing significance[2].

2.2.1 Advantages of Paper

Estimate individual performance: The software is having the capability of estimating and predicting the individual performance of a player in the league games.

2.2.2 Disadvantages of Paper

Neural network: The neural networks can sometimes become very complicated due to a large number of algorithms and data required for it to function and come to a satisfactory result.

2.2.3 How to Overcome the Problems Mentioned in Paper

Combination: To overcome this problem of neural networks, we can combine neural networks with regression for better results in estimating the performance

of the player.

2.3 Twitter Sentiment Analysis Using Hybrid Cuckoo Search Method

Sentiment analysis is one of the prominent fields of data mining that deals with the identification and analysis of sentimental contents generally available at social media. Twitter is one of such social medias used by many users about some topics in the form of tweets[3]. These tweets can be analyzed to find the viewpoints and sentiments of the users by using clustering-based methods. However, due to the subjective nature of the Twitter datasets, metaheuristic-based clustering methods outperforms the traditional methods for sentiment analysis. Therefore, this paper proposes a novel metaheuristic method (CSK) which is based on K-means and cuckoo search. The proposed method has been used to find the optimum cluster-heads from the sentimental contents of Twitter dataset. The efficacy of proposed method has been tested on different Twitter datasets and compared with particle swarm optimization, differential evolution, cuckoo search, improved cuckoo search, gauss-based cuckoo search, and two n-grams methods. Experimental results and statistical analysis validate that the proposed method outperforms the existing methods. The proposed method has theoretical implications for the future research to analyze the data generated through social networks/medias. This method has also very generalized practical implications for designing a system that can provide conclusive reviews on any social issues[3].

2.3.1 Advantages of Paper

Accurate: The results of the sentiments of Twitter-basedareomments is highly accurate using Hybrid Cuckoo search method.

2.3.2 Disadvantages of Paper

Unstructured and grammatical mistakes: The tweets can be highly unstructured and with the presence of grammatical mistakes or typos, it can lead to complications for the software to analyze the tweets.

2.3.3 How to Overcome the Problems Mentioned in Paper

Optimum Cuckoo search: The solution for this is to use the optimum Cuckoo method to solve the highly unstructured tweets and grammatical mistakes.

2.4 Technical Analysis Of Player's Performance

We will go on to identify the best performing footballers in comparison to their teammates. This type of analysis is particularly useful in unveiling the potential of footballers who do not yet play for the most competitive clubs. It also allows us to measure the clubs' dependency on their key players[4].

In the conclusion, we will underline some of the numerous advantages that a well-grounded approach to measure players' technical performance as presented in this report can bring to forward-thinking teams. The CIES Football Observatory research team is at the disposal of professional clubs to help enhance their effectiveness in this field.

2.4.1 Advantages of Paper

Potential of players: It is used to determine the potential of players who will be playing in the upcoming match.

2.4.2 Disadvantages of Paper

Complicated: The analysis of the players' performance can be complex at times due to the many factors involved in the process.

2.4.3 How to Overcome the Problems Mentioned in Paper

KPI: Key Performance Indicator (KPI) can be used to indicate the performances of players and can lead to an easy analysis of the players.

2.5 Technical Review

1. Scrapping :

Scrapping is a good approach in the web system. It is the process of extracting data or information even without the permission of the website owner. In our project we will scrap data from twitter and cricket website sources which is related to the cricket players and data will be scrapped by keywords.

2. Library Used for Scrapping :

- a. Beautiful soup: It is an excellent tool for scraping data or information from web pages. You can use it to scrap tables, images, tweets, public opinions etc. It only scraps information and not the url.

3. Classification:

Classification is based on the tweets we extract from twitter and whether the tweets show a positive, neutral or negative response. For the classification, we will be using the Naive Bayes Algorithm. Naive Bayes Algorithm is mainly used for statistics and probabilities.

2.5.1 Advantages of Technology

- a. It is easily available and free of cost.
- b. It is an open use technology.
- c. It is easy for scrapping and fetching.
- d. Classification accuracy is high in Naive Bayes Algorithm.

2.5.2 Reasons to Use This Technology

- a. We use this technology to fetch information from twitter and cricket website sources.
- b. This technology helps in classification of data. (Naive Bayes Algorithm)

Chapter 3

Project Planning

3.1 Members and Capabilities

Table 3.1: Table of Capabilities

SR. No	Name of Member	Capabilities
1	Shah Shahil Shakir Hussain Hafizunnisa	Python Programming, Testing
2	Sarguroh Junaaid Jawed Sadaf	PHP, Documentation
3	Gavandi Abhay Audumbar Varsha	Python, PHP, Web
4	Syed Areeb Iqbal Ahmad Rehana	Documentation, Web designing

3.2 Roles and Responsibilities

Table 3.2: Table of Responsibilities

SR. No	Name of Member	Role	Responsibilities
1	Shah Shahil Shakir Hussain Hafizunnisa	Team Leader	Designing, Testing, Planning
2	Sarguroh Junaaid Jawed Sadaf	Team Member	Documentation, Coding
3	Gavandi Abhay Audumbar Varsha	Team Member	Coding, Designing
4	Syed Areeb Iqbal Ahmad Rehana	Team Member	Documentation, Planning

3.3 Assumptions and Constraints

3.3.1 Assumptions

The assumption of our project is to assume a data that based on the user knowledge, user experience and useful information is available on hand. We assume that the data we provide is purely true because this purely data is manage and stored online and must be secure from the unauthorized user.

3.3.2 Constraints

In our project, we make schedule for a project to complete on time based on different constraints that required in our project. We may also include the scope of the

project and the cost of the project that required for completing the project. Different quality attributes in projects and resources required in project. No risk tolerance is present in our project.

3.4 Project Management Approach

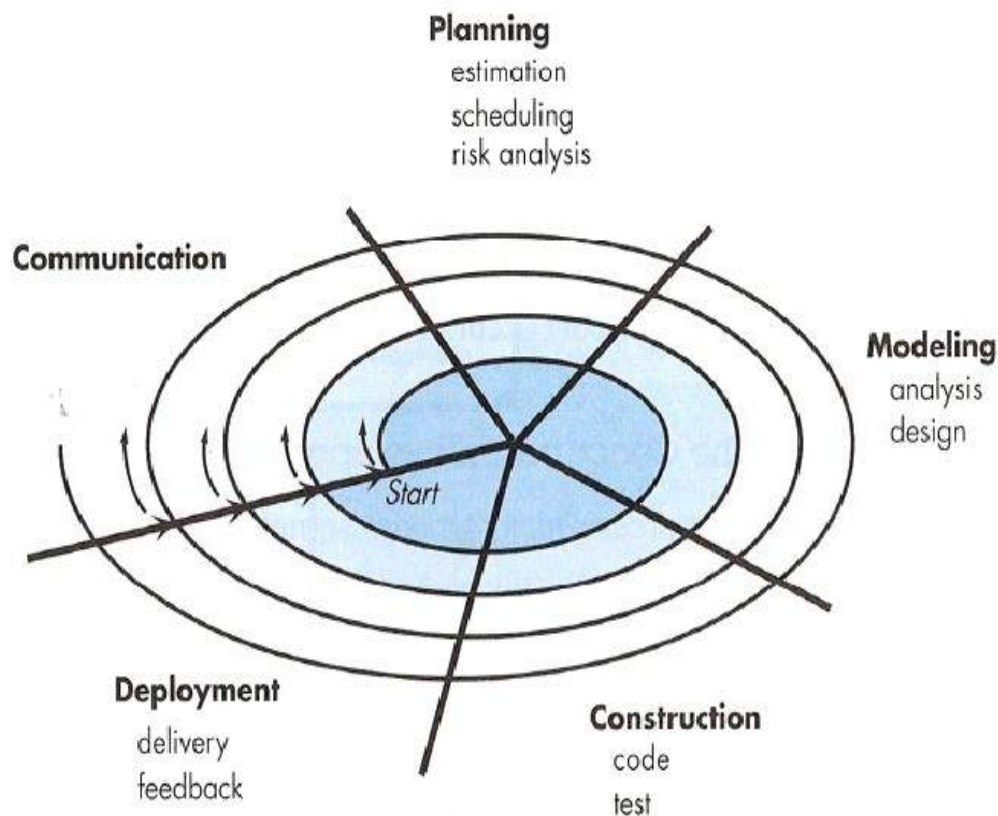


Figure 3.1: Model of our project

In our project we had used spiral model for implementing all the phases successfully. This model involves strategies, which is a combination of incremental and prototype models. This model is suitable for planning and implementing to achieve the goal of the project. It maintains a systematic step wise approach. These are the different phases involved in our project:

3.4.1 Planning

In any project planning phases are most important phase whenever we are going to make any project. So we need to gather proper information related to our project so therefore we had searched different websites which are related to conferences to understand the structure of the websites to scarp.

3.4.2 Risk Analysis

1. Identify the Parsing Structure to scrap.
2. Implementation of fetching function.
3. Identify the Proper Structure to scrap.

3.4.3 Engineering Phase

Testing are also important for any system so before implementation of the project first we have to also test the cases that we are going to implement in our project. We have used Beautiful soup and request library. Once we will integrate these two libraries only parsing part will be remaining that we will get from the website's structure such as HTML tag which is used in website's to built that is about to scrap. once our fetching part will be done then we have to check that we are getting the data from the website's which we have targeted based on the website's tag such as HTML tag. Here for testing purpose we have tested various components of the software. When we are implementing the testing part we successfully get the data from the twitter and cricbuzz that we have targeted to scrap the data such as tweets, ranking.

3.4.4 Evaluation

Vendor's involvement takes place in this Evaluation phase. If vendor wants any specific filter for example a sportsperson for a cream product, so the best sports person for advertisement for cream product will be displayed as a result for the vendor.

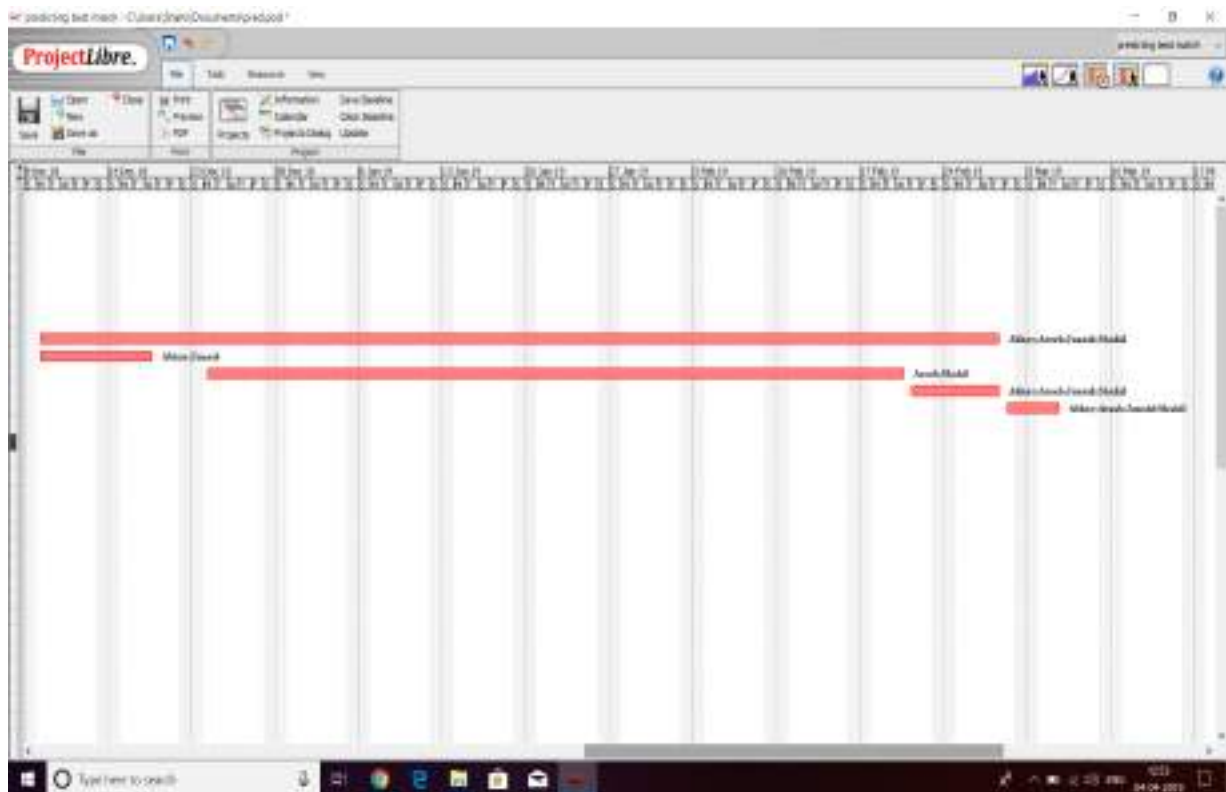
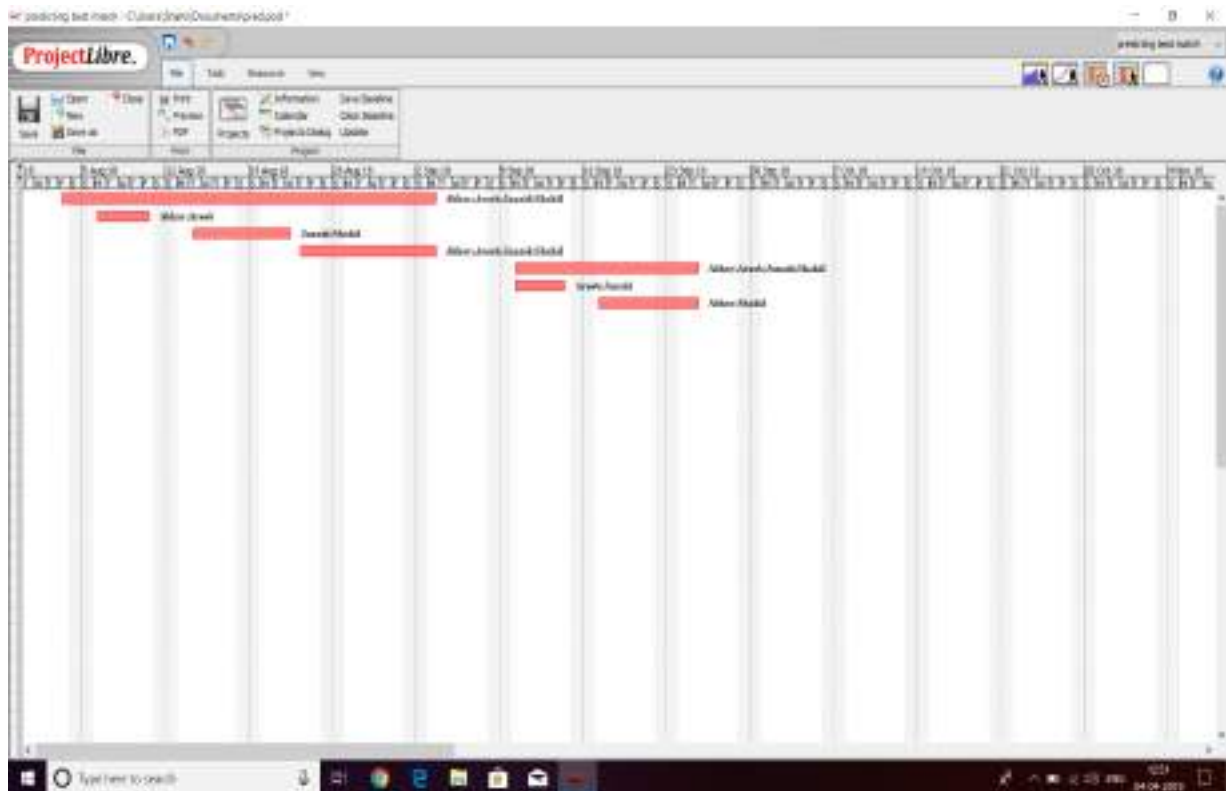
3.5 Ground Rules for the Project

After using our system for choosing the best sports person for product advertisement, our ground rule is that the vendor does not have to go to many websites in search of what player he wants that is related to product advertisement. Instead he can subscribe to our system in order to get full information in a single place, so in our system vendor don't need to go to different website's to get the information.

3.6 Project Budget

1. Beautiful Soup: Free Open Source
2. Request Library: Free Open Source

3.7 Project Timeline



Chapter 4

Software Requirements Specification

4.1 Overall Description

This Software Requirement Specification is the requirement work product that formally shows the vendor the most popular sportsperson for advertisement according to the requirements of the product. The objectives of this document therefore is to formally describe the system's high level requirements including functional requirement, non-functional requirement business rules and constraints.

4.1.1 Product Perspective

The various system tool that have been used in developing the back-end and other tools of the project are being discussed in this section. The back-end is implemented using MySQL which is used to design the database. MySQL is the world second most widely used open source relational database management. The SQL phrase stands for structured query. And PHP is a server side scripting language designed for web development but also used as a general purpose programming language. PHP code is interpreted by a web server with PHP processor module which generates the resulting webpages.

4.1.2 Product Features

The system will provide all the data related to the sports person to the vendor. Depending upon the vendor's role, he/she will be able to access the data related to the product advertisement after searching for trending players. Managing the database by converting them into json file. This made work simple to see the different players data just by providing requirements (For example: suitable for=sports shoes). As a result, the filtered popular face list will be displayed which will only show sports person popular for advertisement of sports shoes.

4.1.3 User Classes and Characteristics

1. Educational Level: At least graduate and should be comfortable with English language.
2. Technical Expertise: Should be a high or middle level employee of the organization comfortable with using general purpose applications on a computer.

4.1.4 Operating Environment

We use the Windows Operating Environment for running the Python software. We use minimum 500GB HardDisk, and we use version of the operating 18.0. We use different software like Spyder.

4.1.5 Design and Implementation Constraints

I Hardware Requirement

- (a) Minimum 500GB space of Hard-Disk.
- (b) Minimum 200MB space of memory.

II Software and Technologies

- (a) MySQL: MySQL is the most popular Open Source Relational SQL Database Man-agement System. MySQL is one of the best RDBMS being used for developing various web-based software applications.
- (b) Python: Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language.

4.2 System Features

4.2.1 System Feature

1. Control Presentation.
2. Creation of data.
3. Organization through web applications.

Description and Priority

The requirements for this feature set describe how the system provides and controls presentation, creation, and organization throughout the Web Application. The system's users are provide information and features related to the conference from which all of their communication with the system will take place. The conference is related to the meeting and perform scraping and sorting on the data.

Stimulus/Response Sequences

- Stimulus : Vendor enters the url of the website.
Response: The system should display the login page of the website to the vendor.
- Stimulus: Vendor enters the username and password for login process.
Response: The website should display main page if the login credentials is correct.
- Stimulus: Vendor searches for a popular sports person and clicks on submit.
Response: A list should be generated and displayed to the vendor.
- Stimulus: Vendor enters requirements for the popular face list generation.
Response: A popular face list should be generated to the vendor.

Functional Requirements

I User Interface

The software provides good graphical user interface for the vendor. It is easy or the user to access. It allows the vendor to generate a popular face list of sportsperson who meet the requirements that he requires for his product advertisement.

II Hardware Interface

- (a) Operating system: Linux
- (b) Hard disk: 40GB
- (c) RAM:256MB
- (d) Processor:Pentium(R)Dual-Core CPU

III Software Interface

- (a) Python language
- (b) MySQL
- (c) Spyder

4.3 External Interface Requirements

4.3.1 User Interface

The Web Server must provide a user interface that will be accessible through any internet browser the major ones being Google Chrome and internet Explorer 12.

4.3.2 Hardware Interface

We don't required any hardware interface in our project. So we required only software interface in our project.

4.3.3 Software Interface

- (a) MySQL: MySQL is the most popular Open Source Relational SQL Database Man-agement System. MySQL is one of the best RDBMS being used for developing var-ious web-based software applications.
- (b) Python: Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language.

4.3.4 Communications Interfaces

Connections to the system will be over TCP/IP connections

4.4 Nonfunctional Requirements

4.4.1 Performance Requirements

The system must be interactive and the delays involved must be less. When we are connecting to the server the delay is because the data is stored or manage online very safely and securely. The data is reliable to the user to see this data very correctly.

4.4.2 Safety Requirements

The data that use for implementation which concerned with the possible loss or harmful used of the data. The data is stored online is very secure because these data is access by only authorized user by providing username and password to the we-bapp. The external policies and safety issue that the product design must be satisfied.

4.4.3 Security Requirements

The server on which the Online Data is stored will have its own security to prevent unauthorized write/delete access. There is no restriction on read access. The use of

email by an Author or Reviewer is on the client systems and thus is external to the system. The PC on which the database resides will have its own security. Only the Editor will have physical access to the machine and the program on it.

Chapter 5

System Design

5.1 System Requirements Definition

We have made a system which will scrap the data from twitter and cricbuzz and store it into the database then it will analyze the data as per user requirement based on requirement of the vendor, filter it and display the popular face list of sports person to the vendor. So once system will get online it will scrap the tweets and rankings of sports personalities if there will be any entry comes into the database. If any new entry or data does not come or any duplicate entry will come so system will automatically skip the data and will not store any data which has been already scrapped and stored into the database. We have made the system in python language and also used some python libraries which are suitable to scrap the data from the websites so this system will be beneficial for those who search the information that are related to conference from different difference websites so for that types of user there are no need to go to the different websites to collect the information .Our system will automatically scrap the data and display it at one place.

5.1.1 Functional Requirements

1. Fetching: It plays a significant role in our project. We have used Beautiful soup which scraps the data in the system.
2. Classification: Once the data is fetched, we then classify whether the data received is positive, negative or neutral.
3. Storage: Store function is very essential to store the data.In our project it will store the data into the database.

Use-case Diagram

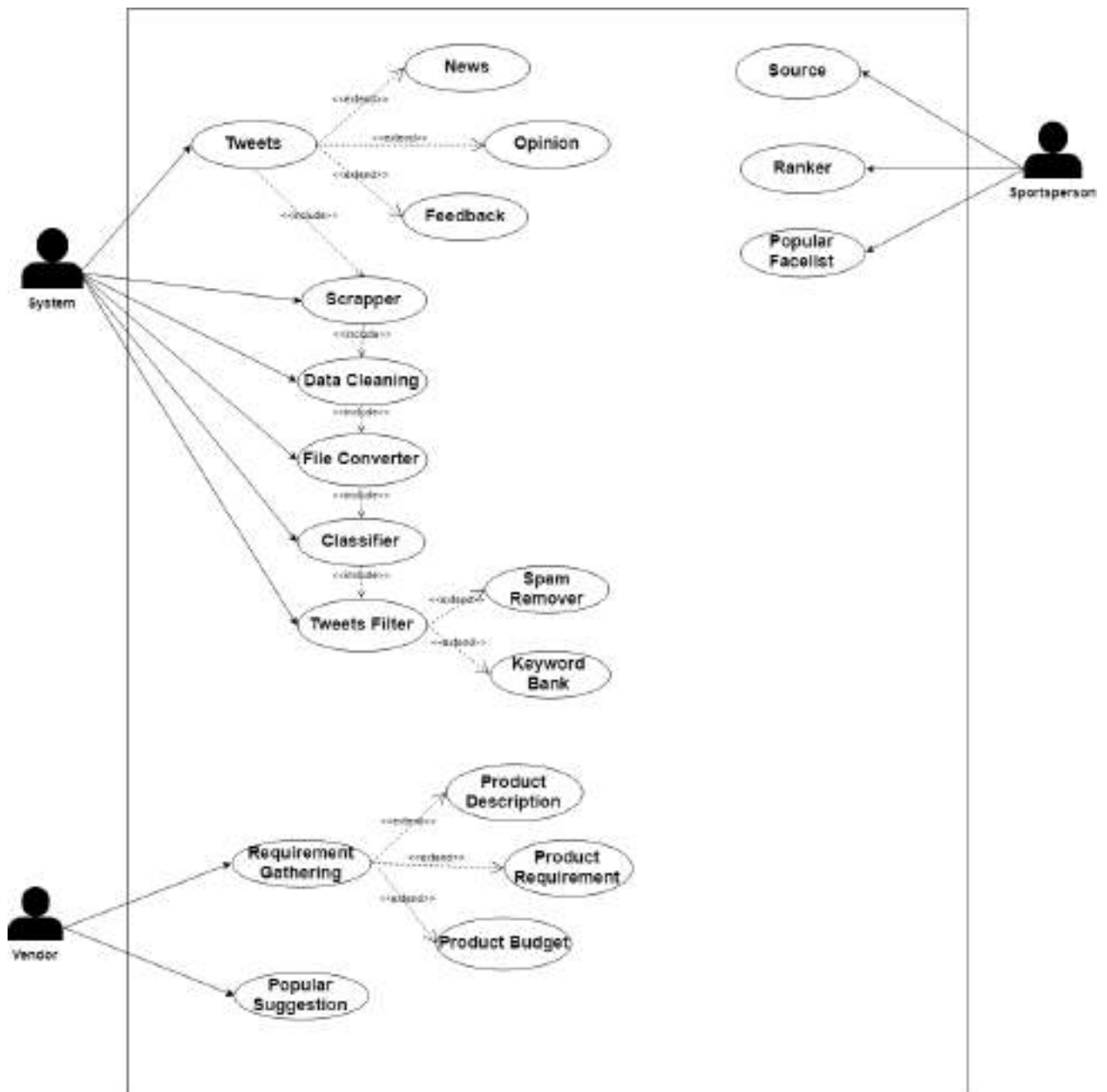


Figure 5.1: Use Case

There are 3 actors in our use case diagram. They are the vendor, system and the sportsperson. The system's work is to scrap the data from sources such as Twitter and cricbuzz, clean the data which is fetched and convert it into a json file. After conversion of file into json format, the task of the system is to classify the data and filter it. The system has to remove spams as well. The sportsperson is fetched from source such as cricbuzz, which allows us to get the rank of the sportsperson. We generate a popular facelist according to the sentiment of the people. The job of the vendor is to search for the suitable player for the product and get a facelist. However, the vendor can filter the list of sports person according to the needs of the product.

A final list of the popular sports person will be generated according to the vendor's requirements.

Data-flow Diagram

DFD Level 0 :

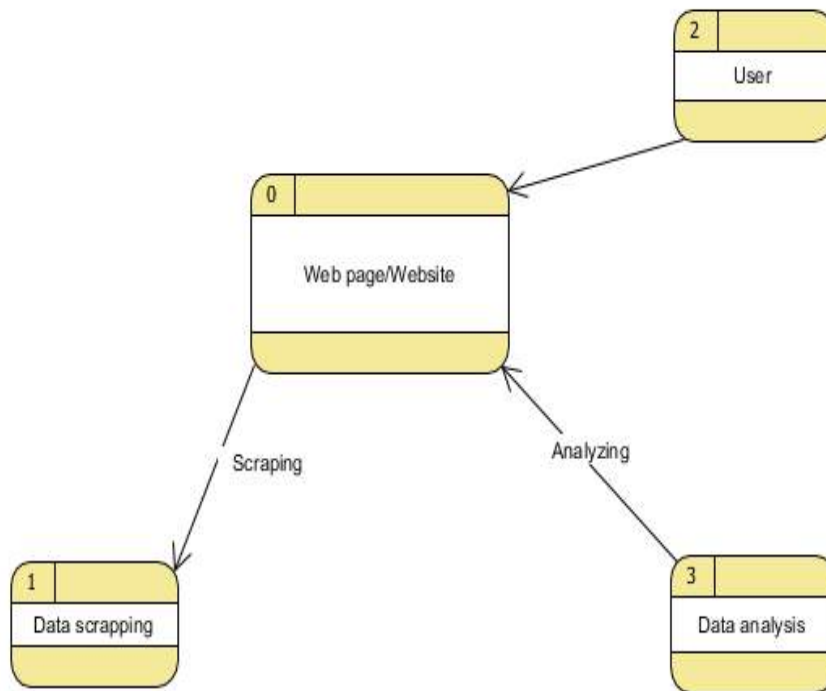


Figure 5.2: DFD Level 0

It contains four total no. of process in our DFD level 0 diagram. It has website, user, data scrapping, data analysis. These are the part of the process in our level 0 DFD diagram. Firstly the authenticated user, has to login to the website. Further the scrapping is done from various sources such as twitter and cricbuzz, and the required data is analyzed to achieve the requirement of the user/vendor.

DFD Level 1 :

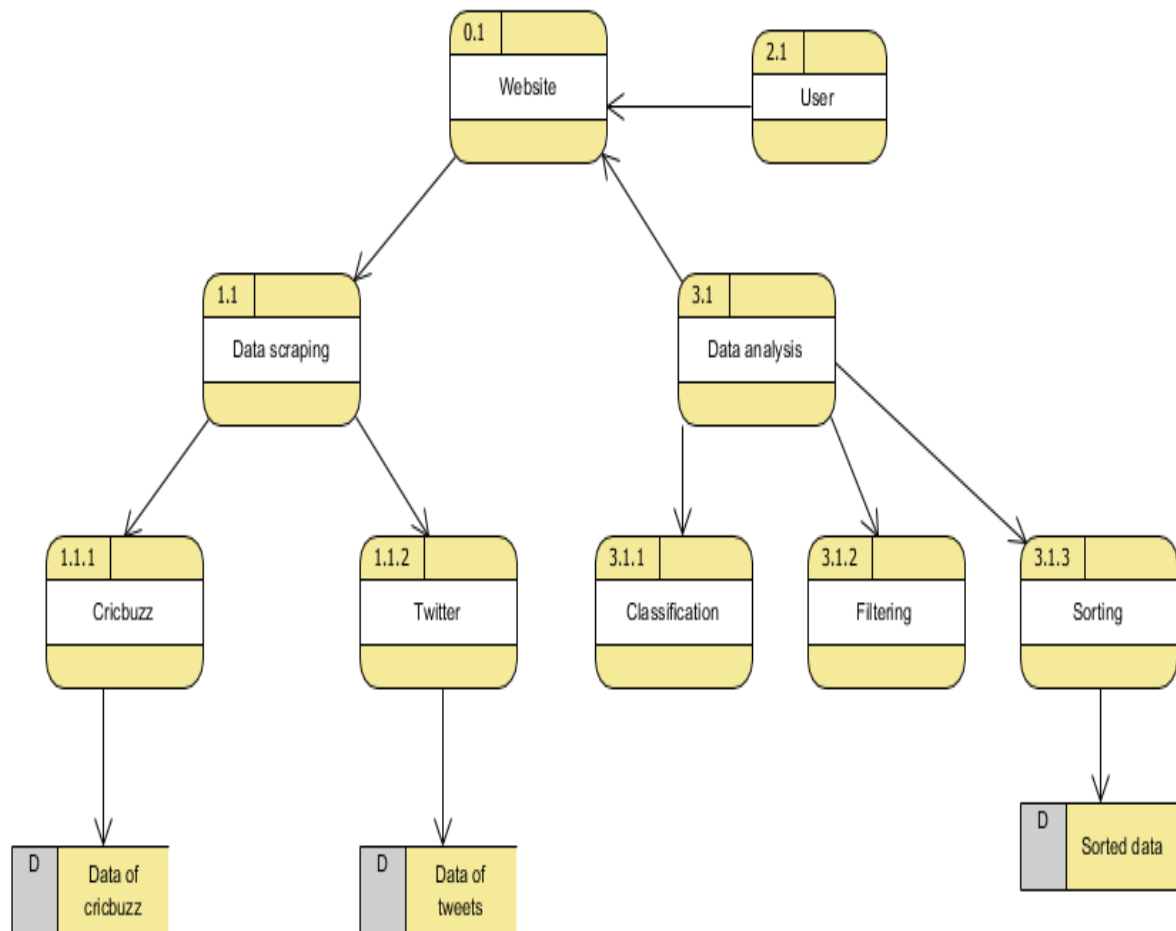


Figure 5.3: DFD Level 1

In DFD level 1, the level 0 is enhanced into a greater extent to show the proper clarification and the data flow of the project. In this, the above process is enhanced such as the data analysis is done on the basis of classification, filtering and sorting of the data. After this the required data is stored into the database. Data scraping is done on the fundamental approaches which focuses on major process such as twitter and cricbuzz.

DFD Level 2 :

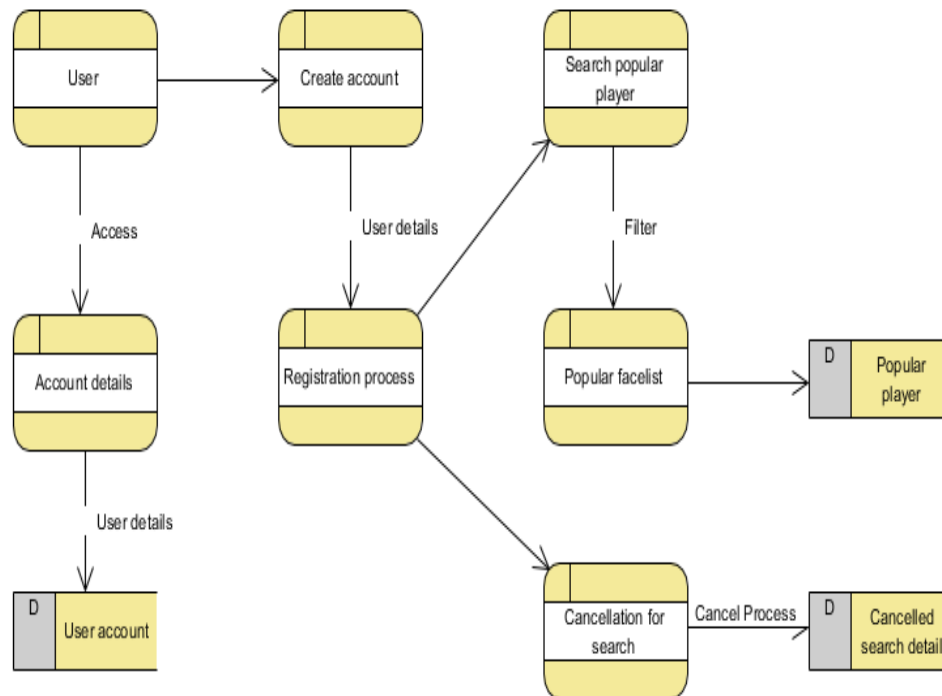


Figure 5.4: DFD Level 2

In DFD level 2, the user process is enhanced. It contains two types of users: Authenticated and Unauthenticated user. The authenticated user can login and they can access our website and search for the players who can be suitable for their product advertisement. The unauthenticated user does not have the right to access the popular facelist through the website. It is mandatory for them to register and become an authenticated user. All the activity of the user is stored in the database.

5.1.2 System Requirements (non-functional requirements)

We have made the system in python language and also used some tools which are suitable for scraping the data from various sources, so this system will be beneficial for those who search for players for product advertisement. The system must be interactive and the delays involved must be less. The data is stored online which is very secure because the data can be accessed by only authorized user.

Database Schema/ E-R Diagram

5.2 System Architecture Design

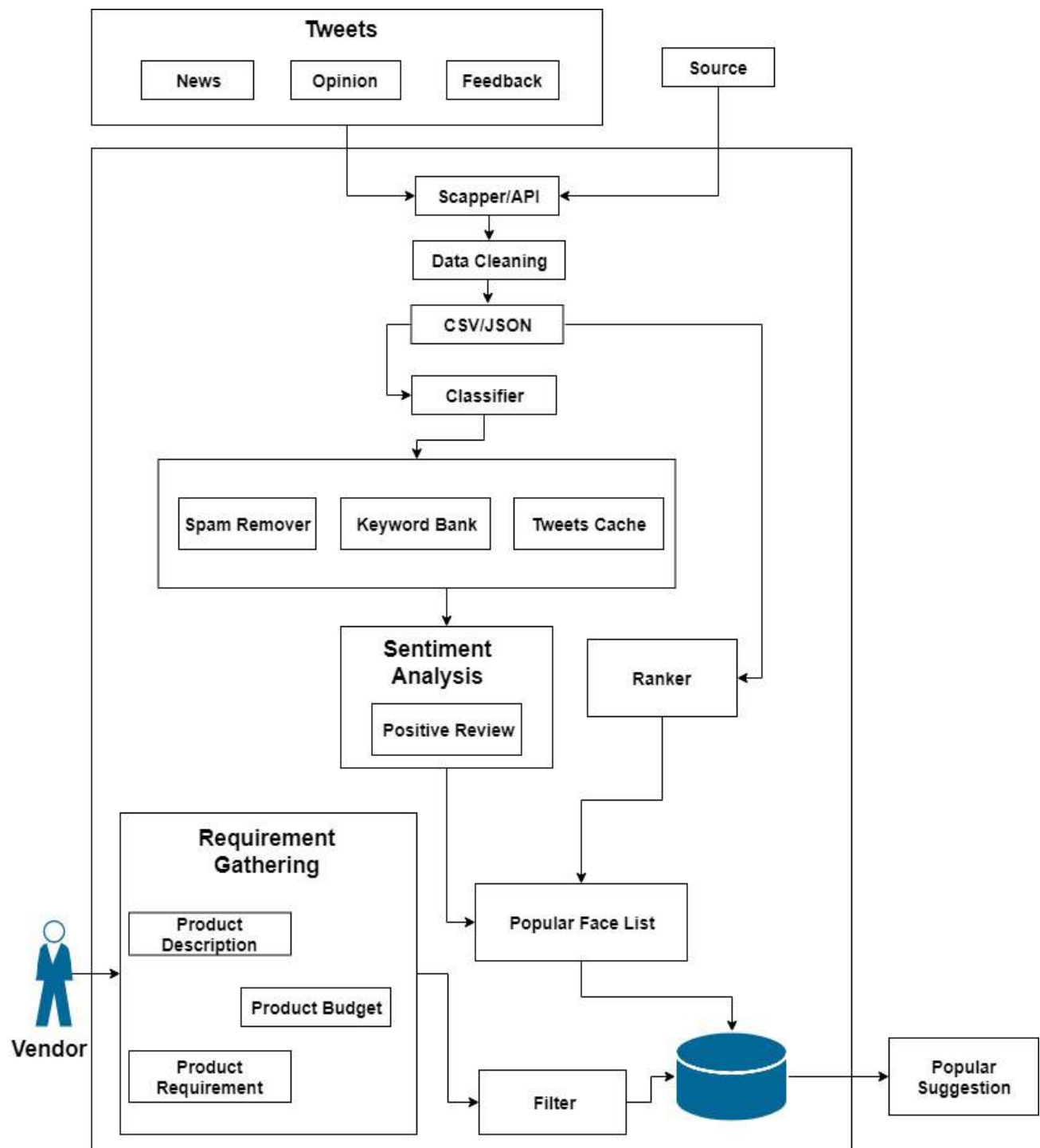


Figure 5.5: System Architecture

Our system is based on the prediction of popular sportsman facelist. In the system architecture the first step is to extract the data/ scrap the data from twitter and cricbuzz. Through twitter we are scraping tweets and from cricbuzz we are scrap-

ing the rank of the players. Scraping of the data is done by beautiful soup. After the scraping, the data is cleaned and all the unnecessary data is thrown out and the cleaned data is stored. After that the data is converted into a json file format for utilization of the data. The classifier segregates data into spam, keywords and cache. There is a keyword bank which is used to train the system using the acquired data. Training of data is done through keyword bank. The next process is to do the sentimental analysis on the tweets we fetched and to take only the percentage of positive sentiment of the player. The positive sentiment percentage and the rank are clubbed together and a popular face list is generated. This popular face list is displayed to the user which he can view. The vendor can apply certain specific requirements according to the product and generate a new popular face list which suits the product's requirements.

5.3 Sub-system Development

In our project we will scrap the data or information from different websites or sources which is related to cricket players. For the scrapping of the data, we are using a tool called Beautiful soup. Also, there is a need to classify the data which has been scrapped and for this we are using the Naive Bayes Algorithm. In our sub-system we are using 3 modules.

5.3.1 Fetching Data

Firstly, we would start scraping the data from various websites or sources. Now the scraped data from twitter will be in the form of tweets, and these tweets can be further classified into news, opinions, feedback. After we have the data with us, we can fetch the data using beautiful soup.

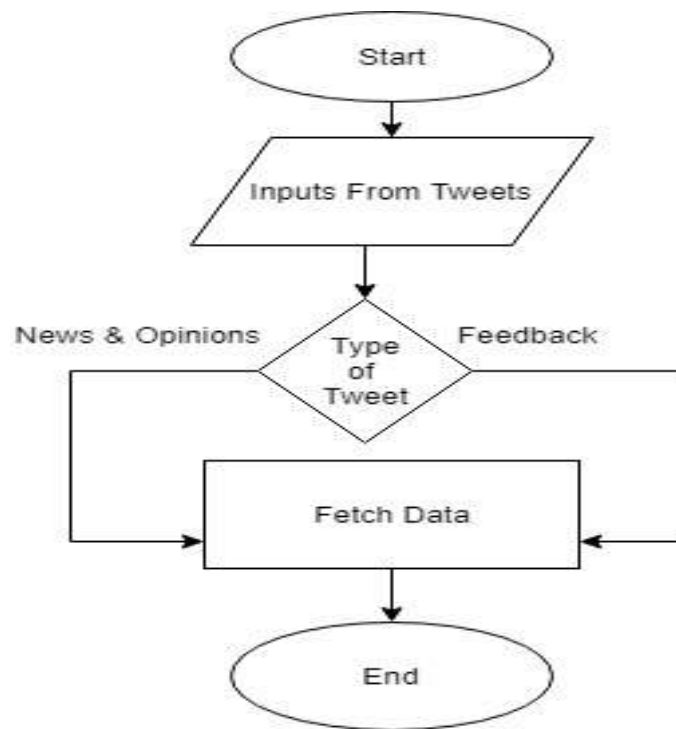


Figure 5.6: Fetching data

5.3.2 Dataset

After the data is scrapped, the data is cleaned and converted into a json file for usage. The next step is to train the system to verify whether the tweet is positive, negative or neutral. To do this, we have a keyword bank which will train itself in order to classify whether a tweet is positive, negative or neutral. We then take the rank of the cricket player and the positive tweets for a player and club them together to generate a list of popular facelist.

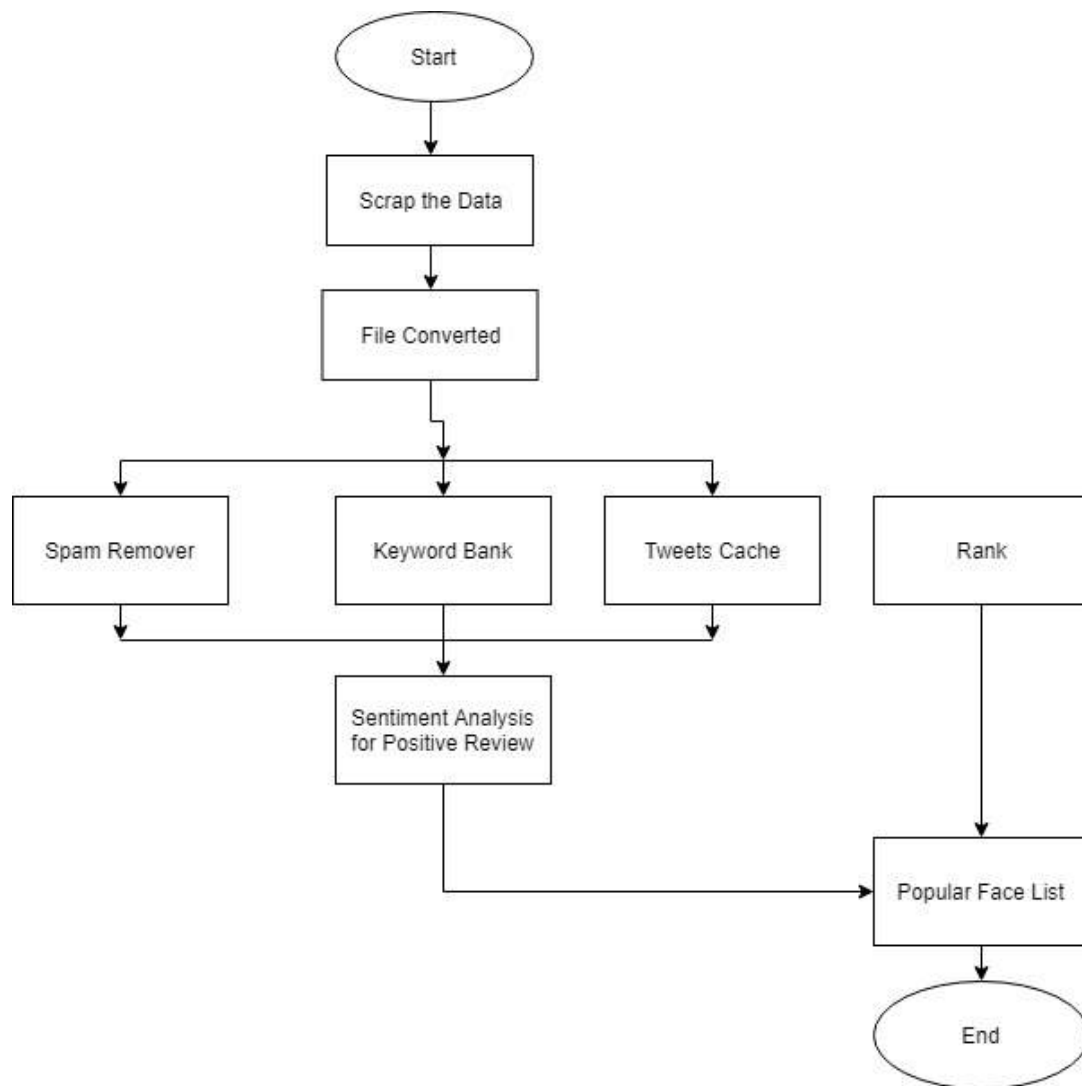


Figure 5.7: Dataset

5.3.3 Filtering of Data

In this module, the vendor plays a very pivotal role in the output. The requirements is taken from the vendor according to the product i.e budget input,product requirement,description of product. The popular facelist which was generated before is now filtered according to the vendors requirements and a new suggested popular facelist of sports person is generated and displayed to the user.

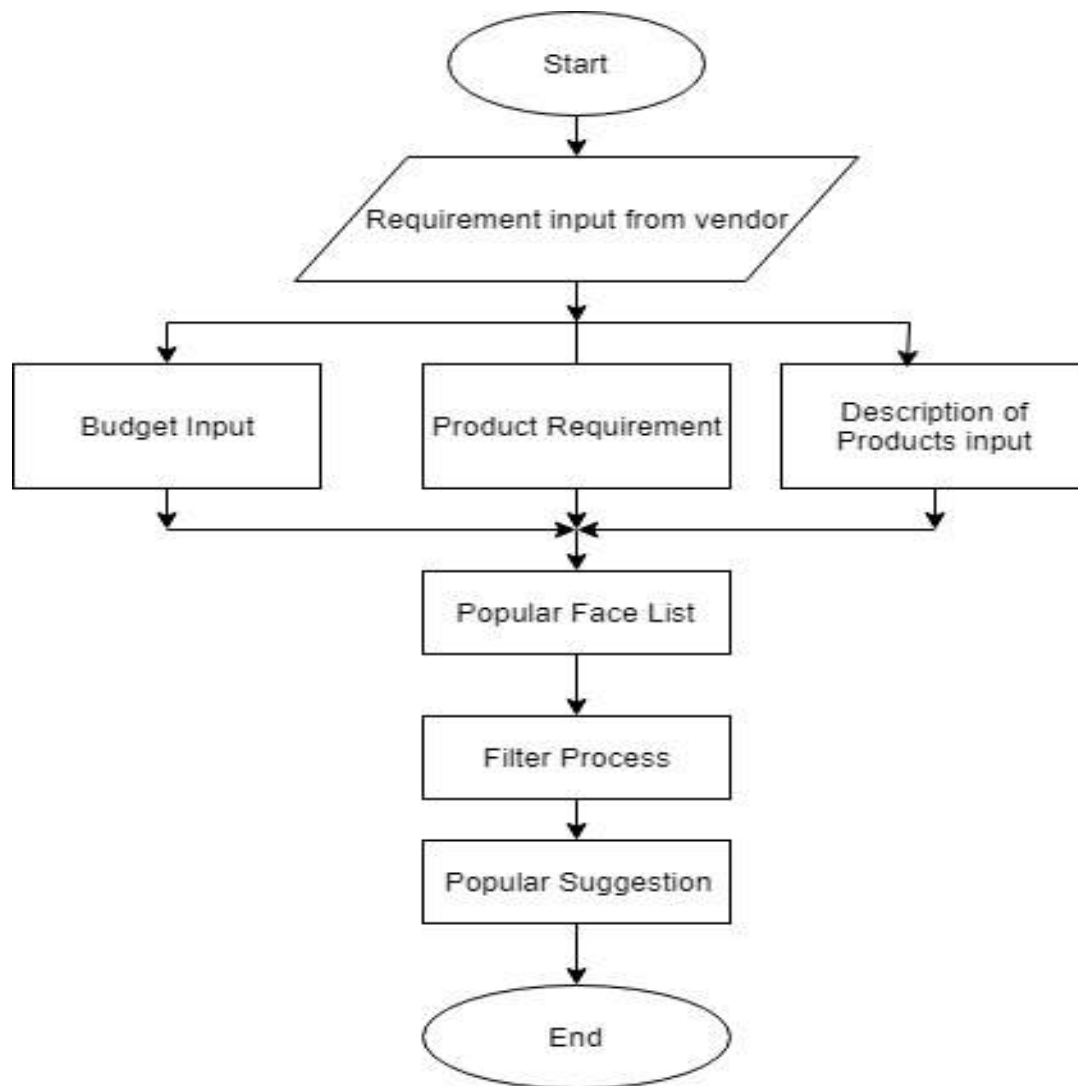


Figure 5.8: Filtering of data

5.4 Systems Integration

Our framework depends on the forecast of the famous sportsman face list. In the framework engineering the initial step is to remove the information/scrap the information from Twitter and cricbuzz. Through Twitter, we are scraping tweets and from cricbuzz, we are scraping the position of the players. Scraping of the information is finished by beautiful soup. After the scraping, the information is cleaned and all the pointless information is tossed out and the cleaned information is put away. After that, the information is changed over into a JSON document design for use of the information. The classifier isolates information into spam, catchphrases and store. There is a keyword bank which is utilized to prepare the framework utilizing the gained information. Preparing of information is done through keyword bank. The following procedure is to do the nostalgic investigation on the tweets we brought and to take just the level of positive estimation of the player. The positive assessment rate and the rank are clubbed together and a famous face list is produced.

This mainstream face list is shown to the client which he can see. The vendor/user can apply certain particular prerequisites as indicated by the item and create another famous face list which suits the item's necessities.

5.4.1 Class Diagram

Classes have various entity and relation with each other. Like we can say one class depends on another entity and relation to make proper data flow. The class diagram also defines the functionality of a particular class or we can say a particular class can do action on various function layer.

Real World Class Perspective:

A vendor can upload n number of product so the system can generate list according to product requirement after that list will suggest particular sportsperson which is suitable for the product advertisement according to its popularity and ranking.

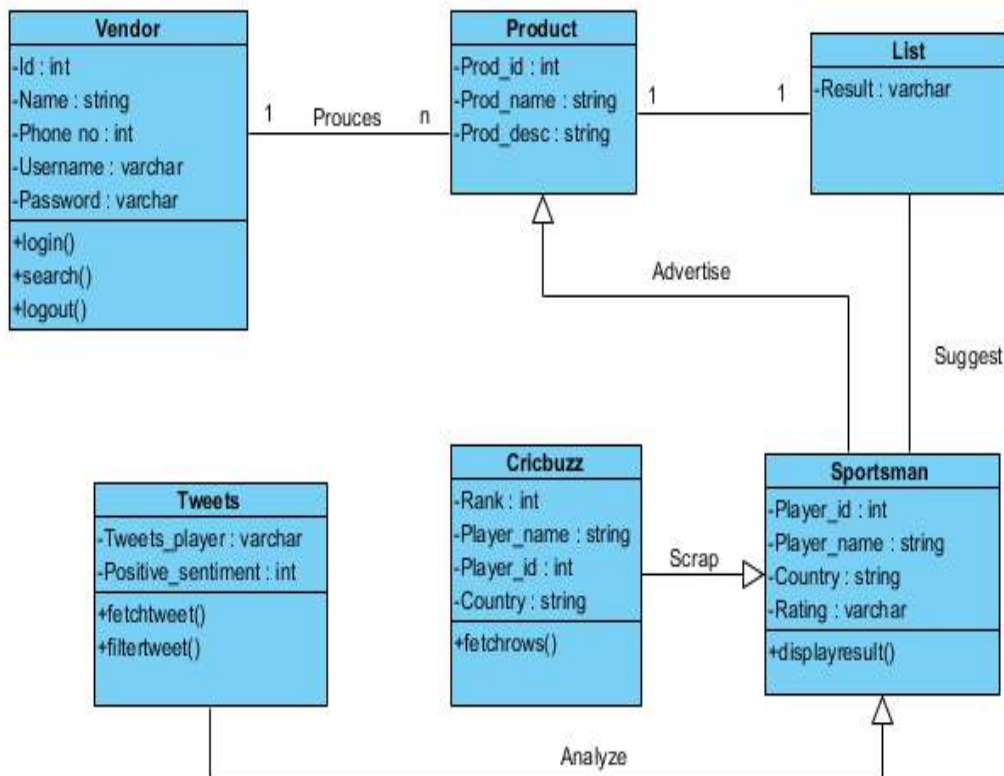


Figure 5.9: Class Diagram

5.4.2 Sequence Diagram

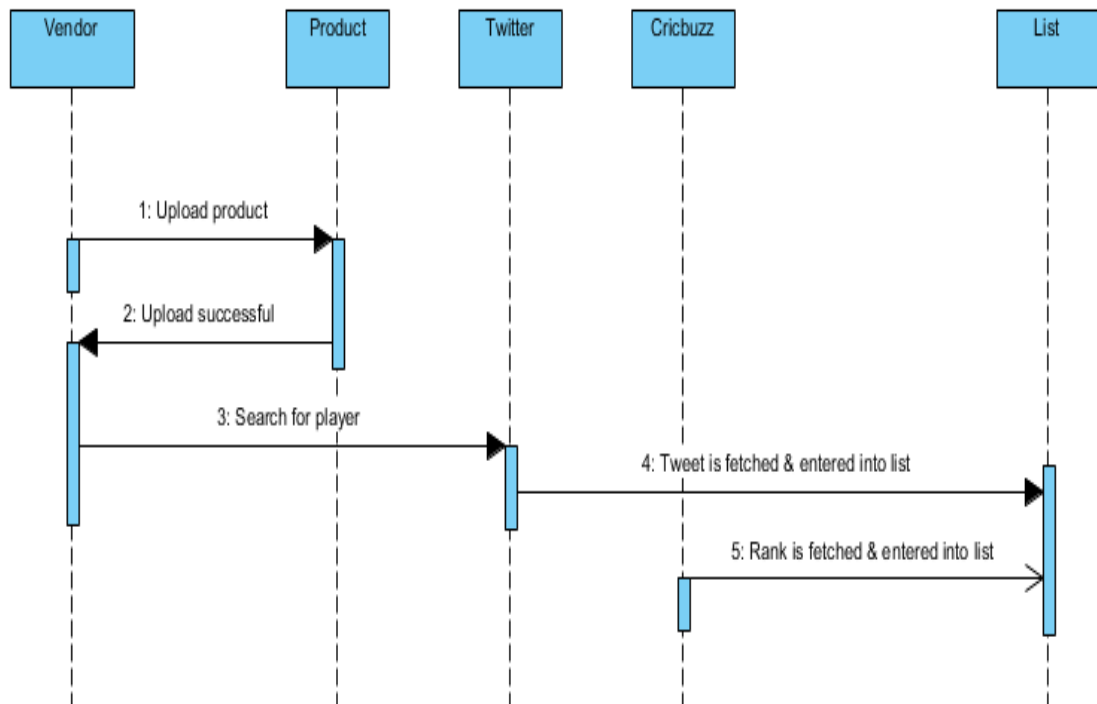


Figure 5.10: Sequence diagram stage 1

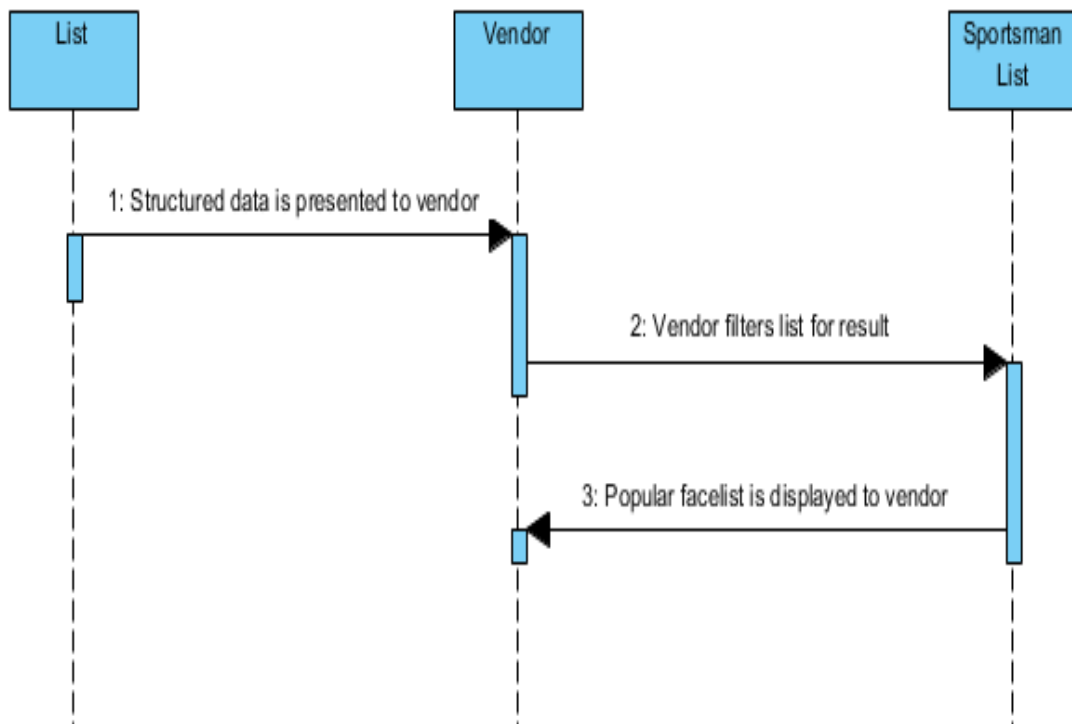


Figure 5.11: Sequence diagram stage 2

There are 7 lifelines in our system in sequence diagram. The vendor will upload a product. Twitter and Cricbuzz data's will be fetched and entered into a list of sorted and cleaned data. The structured data is presented to the vendor in the form of a list which contains the suggested trending cricket players. Vendor then enters the requirements according to the product and the list is filtered. The filtered list is the popular face list which is displayed on the web page to the vendor according to the requirements of the product.

5.4.3 Component Diagram

Data scraper has a major function for the extraction of data. It can be done on various websites as in our case it is twitter and cricbuzz. With the help of data scraper, scraping is done and the scraped data is stored. Further the data is classified into positive,negative and neutral sentiments using the classifier. And then the filtered result is displayed to the user which is done by the filter component.

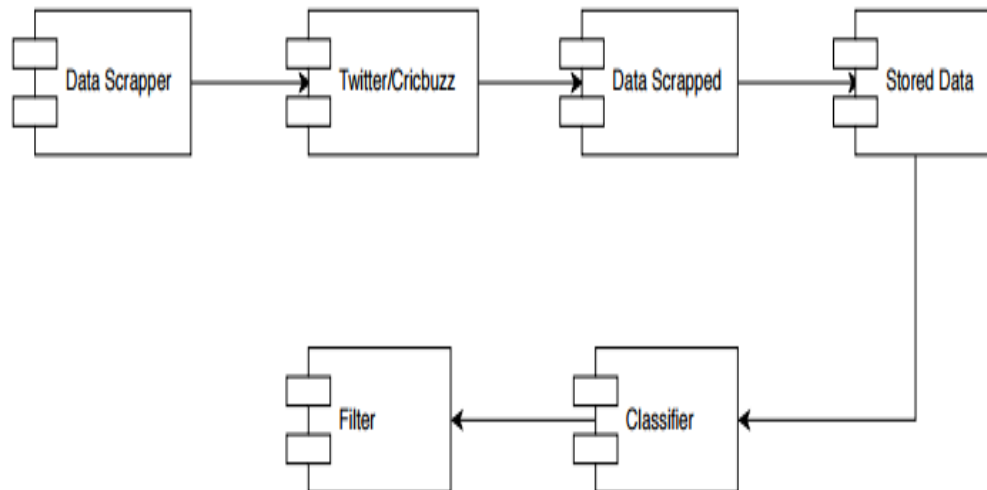


Figure 5.12: Component Diagram

5.4.4 Deployment Diagram

From the different various sources and websites such as twitter and cricbuzz, the data is scraped by the data scraper. Later on the data is further analyzed/classified by the data analyzer. The resulted data is then stored in the database server.

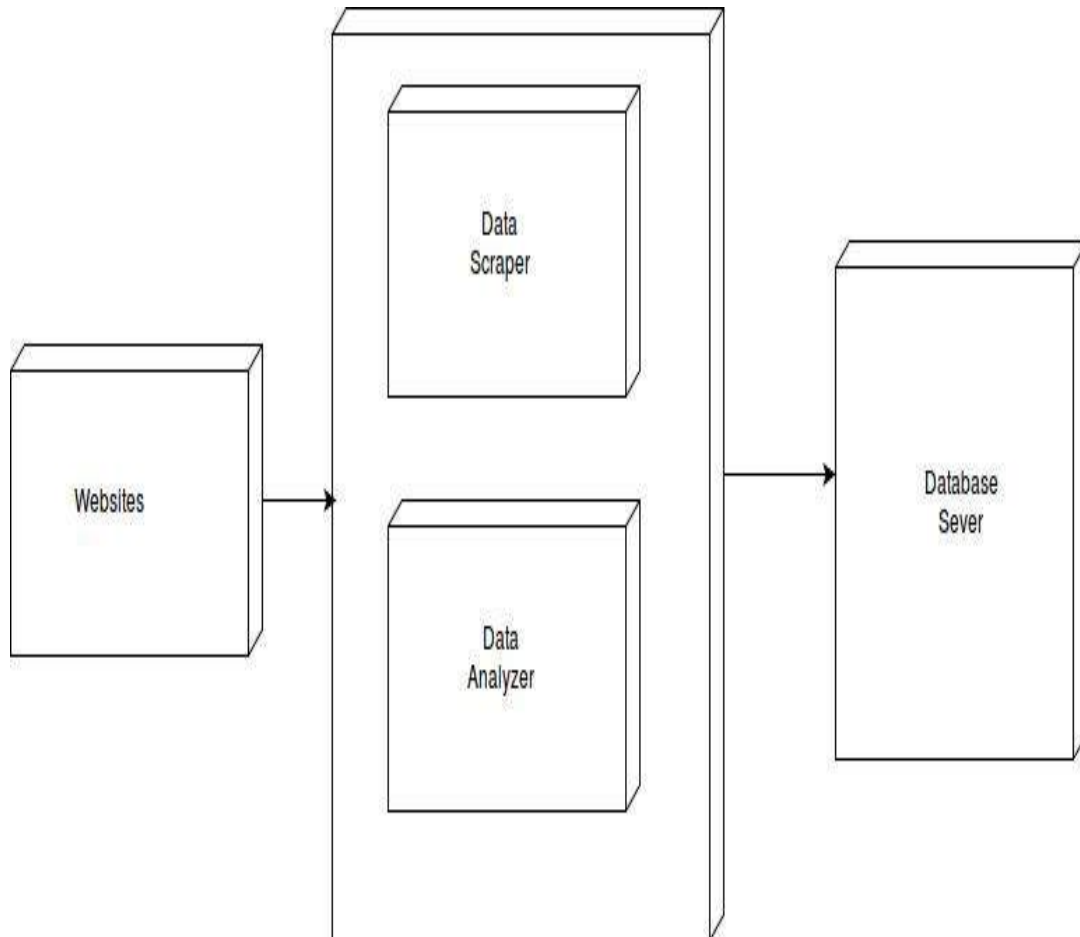


Figure 5.13: Deployment Diagram

Chapter 6

Implementation

6.1 Scrapping From Cricbuzz



```

1: 2018 Microsoft Corporation. All rights reserved.
2: (LibraryPath\Python\Scripts\python.exe)
3:
4: C:\Users\Shubham> python script.py
5:
6:
7:
8:
9:
10:
11:
12:
13:
14:
15:
16:
17:
18:
19:
20:
21:
22:
23:
24:
25:
26:
27:
28:
29:
30:
31:
32:
33:
34:
35:
36:
37:
38:
39:
40:
41:
42:
43:
44:
45:
46:
47:
48:
49:
50:
51:
52:
53:
54:
55:
56:
57:
58:
59:
60:
61:
62:
63:
64:
65:
66:
67:
68:
69:
70:
71:
72:
73:
74:
75:
76:
77:
78:
79:
80:
81:
82:
83:
84:
85:
86:
87:
88:
89:
90:
91:
92:
93:
94:
95:
96:
97:
98:
99:
100:
101:
102:
103:
104:
105:
106:
107:
108:
109:
110:
111:
112:
113:
114:
115:
116:
117:
118:
119:
120:
121:
122:
123:
124:
125:
126:
127:
128:
129:
130:
131:
132:
133:
134:
135:
136:
137:
138:
139:
140:
141:
142:
143:
144:
145:
146:
147:
148:
149:
150:
151:
152:
153:
154:
155:
156:
157:
158:
159:
160:
161:
162:
163:
164:
165:
166:
167:
168:
169:
170:
171:
172:
173:
174:
175:
176:
177:
178:
179:
180:
181:
182:
183:
184:
185:
186:
187:
188:
189:
190:
191:
192:
193:
194:
195:
196:
197:
198:
199:
200:
201:
202:
203:
204:
205:
206:
207:
208:
209:
210:
211:
212:
213:
214:
215:
216:
217:
218:
219:
220:
221:
222:
223:
224:
225:
226:
227:
228:
229:
230:
231:
232:
233:
234:
235:
236:
237:
238:
239:
240:
241:
242:
243:
244:
245:
246:
247:
248:
249:
250:
251:
252:
253:
254:
255:
256:
257:
258:
259:
260:
261:
262:
263:
264:
265:
266:
267:
268:
269:
270:
271:
272:
273:
274:
275:
276:
277:
278:
279:
280:
281:
282:
283:
284:
285:
286:
287:
288:
289:
290:
291:
292:
293:
294:
295:
296:
297:
298:
299:
300:
301:
302:
303:
304:
305:
306:
307:
308:
309:
310:
311:
312:
313:
314:
315:
316:
317:
318:
319:
320:
321:
322:
323:
324:
325:
326:
327:
328:
329:
330:
331:
332:
333:
334:
335:
336:
337:
338:
339:
340:
341:
342:
343:
344:
345:
346:
347:
348:
349:
350:
351:
352:
353:
354:
355:
356:
357:
358:
359:
360:
361:
362:
363:
364:
365:
366:
367:
368:
369:
370:
371:
372:
373:
374:
375:
376:
377:
378:
379:
380:
381:
382:
383:
384:
385:
386:
387:
388:
389:
390:
391:
392:
393:
394:
395:
396:
397:
398:
399:
400:
401:
402:
403:
404:
405:
406:
407:
408:
409:
410:
411:
412:
413:
414:
415:
416:
417:
418:
419:
420:
421:
422:
423:
424:
425:
426:
427:
428:
429:
430:
431:
432:
433:
434:
435:
436:
437:
438:
439:
440:
441:
442:
443:
444:
445:
446:
447:
448:
449:
450:
451:
452:
453:
454:
455:
456:
457:
458:
459:
460:
461:
462:
463:
464:
465:
466:
467:
468:
469:
470:
471:
472:
473:
474:
475:
476:
477:
478:
479:
480:
481:
482:
483:
484:
485:
486:
487:
488:
489:
490:
491:
492:
493:
494:
495:
496:
497:
498:
499:
500:
501:
502:
503:
504:
505:
506:
507:
508:
509:
510:
511:
512:
513:
514:
515:
516:
517:
518:
519:
520:
521:
522:
523:
524:
525:
526:
527:
528:
529:
530:
531:
532:
533:
534:
535:
536:
537:
538:
539:
540:
541:
542:
543:
544:
545:
546:
547:
548:
549:
550:
551:
552:
553:
554:
555:
556:
557:
558:
559:
560:
561:
562:
563:
564:
565:
566:
567:
568:
569:
570:
571:
572:
573:
574:
575:
576:
577:
578:
579:
580:
581:
582:
583:
584:
585:
586:
587:
588:
589:
590:
591:
592:
593:
594:
595:
596:
597:
598:
599:
600:
601:
602:
603:
604:
605:
606:
607:
608:
609:
610:
611:
612:
613:
614:
615:
616:
617:
618:
619:
620:
621:
622:
623:
624:
625:
626:
627:
628:
629:
630:
631:
632:
633:
634:
635:
636:
637:
638:
639:
640:
641:
642:
643:
644:
645:
646:
647:
648:
649:
650:
651:
652:
653:
654:
655:
656:
657:
658:
659:
660:
661:
662:
663:
664:
665:
666:
667:
668:
669:
670:
671:
672:
673:
674:
675:
676:
677:
678:
679:
680:
681:
682:
683:
684:
685:
686:
687:
688:
689:
690:
691:
692:
693:
694:
695:
696:
697:
698:
699:
700:
701:
702:
703:
704:
705:
706:
707:
708:
709:
710:
711:
712:
713:
714:
715:
716:
717:
718:
719:
720:
721:
722:
723:
724:
725:
726:
727:
728:
729:
730:
731:
732:
733:
734:
735:
736:
737:
738:
739:
740:
741:
742:
743:
744:
745:
746:
747:
748:
749:
750:
751:
752:
753:
754:
755:
756:
757:
758:
759:
760:
761:
762:
763:
764:
765:
766:
767:
768:
769:
770:
771:
772:
773:
774:
775:
776:
777:
778:
779:
780:
781:
782:
783:
784:
785:
786:
787:
788:
789:
790:
791:
792:
793:
794:
795:
796:
797:
798:
799:
800:
801:
802:
803:
804:
805:
806:
807:
808:
809:
810:
811:
812:
813:
814:
815:
816:
817:
818:
819:
820:
821:
822:
823:
824:
825:
826:
827:
828:
829:
830:
831:
832:
833:
834:
835:
836:
837:
838:
839:
840:
841:
842:
843:
844:
845:
846:
847:
848:
849:
850:
851:
852:
853:
854:
855:
856:
857:
858:
859:
860:
861:
862:
863:
864:
865:
866:
867:
868:
869:
870:
871:
872:
873:
874:
875:
876:
877:
878:
879:
880:
881:
882:
883:
884:
885:
886:
887:
888:
889:
890:
891:
892:
893:
894:
895:
896:
897:
898:
899:
900:
901:
902:
903:
904:
905:
906:
907:
908:
909:
910:
911:
912:
913:
914:
915:
916:
917:
918:
919:
920:
921:
922:
923:
924:
925:
926:
927:
928:
929:
930:
931:
932:
933:
934:
935:
936:
937:
938:
939:
940:
941:
942:
943:
944:
945:
946:
947:
948:
949:
950:
951:
952:
953:
954:
955:
956:
957:
958:
959:
960:
961:
962:
963:
964:
965:
966:
967:
968:
969:
970:
971:
972:
973:
974:
975:
976:
977:
978:
979:
980:
981:
982:
983:
984:
985:
986:
987:
988:
989:
990:
991:
992:
993:
994:
995:
996:
997:
998:
999:
1000:

```

Figure 6.1: Scrapping script from Cricbuzz

```

1 import requests
2 from bs4 import BeautifulSoup
3 import csv
4 import pandas as pd
5 import numpy
6 req = requests.get("https://www.cricbuzz.com/cricket-stats/icc-rankings/men/
    batting")
7 #print(req.text)
8 #content = dir(BeautifulSoup)
9 #print(content)
10 soup = BeautifulSoup(req.text, 'lxml')
11 #print(soup.prettify())

```

```

12 lp = list();
13 lplayer = list();
14 lrating = list();
15 lcountry = list();
16 final_scrap = list();
17 position = soup.find_all("div",class_="cb-col cb-col-16 cb-rank-tbl cb-font-16")
18 player = soup.find_all("a",class_="text-hvr-underline text-bold cb-font-16")
19 rating = soup.find_all("div",class_="cb-col cb-col-17 cb-rank-tbl pull-right")
20 country = soup.find_all("div",class_="cb-font-12 text-gray")
21 for pos in position:
22     lp.append(pos.text)
23 for pla in player:
24     lplayer.append(pla.text)
25 for rate in rating:
26     lrating.append(rate.text)
27 for countr in country:
28     lcountry.append(countr.text)
29
30
31 #Fetch batting test
32 with open('batting_test.csv','w') as csv_file:
33     fieldnames = ['Position','Player','Rating','Type','Type_Player','Country']
34     writer = csv.DictWriter(csv_file,fieldnames=fieldnames)
35     type = 'batting_test'
36     writer.writeheader()
37     for i in range(0,100):
38         writer.writerow({'Position':lp[i],'Player':lplayer[i],'Rating':lrating[i],
39                             'Type':type,'Country':lcountry[i]})
39     csv_file.close();
40
41 #Fetch batting odi
42 with open('batting_odi.csv','w') as csv_file:
43     fieldnames = ['Position','Player','Rating','Type','Country']
44     writer = csv.DictWriter(csv_file,fieldnames=fieldnames)
45     type = 'batting_odi'
46     writer.writeheader()
47     for i in range(100,200):
48         writer.writerow({'Position':lp[i],'Player':lplayer[i],'Rating':lrating[i],
49                             'Type':type,'Country':lcountry[i]})
49     csv_file.close();
50
51 #Fetch batting t20
52 with open('batting_t20.csv','w') as csv_file:
53     fieldnames = ['Position','Player','Rating','Type','Country']
54     writer = csv.DictWriter(csv_file,fieldnames=fieldnames)
55     type = 'batting_t20'
56     writer.writeheader()
57     for i in range(200,299):
58         writer.writerow({'Position':lp[i],'Player':lplayer[i],'Rating':lrating[i],
59                             'Type':type,'Country':lcountry[i]})
59     csv_file.close();

```



```
1  from textblob import TextBlob
2  import sys, tweepy
3  import matplotlib.pyplot as plt
4  import mysql.connector
5
6
7  def percentage(part, whole):
8      return 100*float(part)/float(whole)
9
10 consumerKey="ZeGyqaUBzYExcO4MHClDFBaMI"
11 consumerSecret="edb3zFTHtjOLNzuaR3HUIsTyBR05M01UMaxkqJhuGQsF1fVn55"
12 accessToken="1098235662753398784-2177Cg9fViEcJduSVTxEaxeoKdGZYv"
13 accessTokenSecret="bqXBx4LiiviNy5euaNel8nFPlbrbQwottD8DXcBtKQ8Cx"
14
15 auth = tweepy.OAuthHandler(consumerKey, consumerSecret)
16 auth.set_access_token(accessToken, accessTokenSecret)
17 api = tweepy.API(auth)
18
19
20 searchTerm = input("enter keyword/hashtag to search: ")
21 noOfSearchTerms = int(input("how many tweets to analyze: "))
22
23 tweets = tweepy.Cursor(api.search, q=searchTerm, lang="en").items(
24     noOfSearchTerms)
25
26 positive = 0
27 negative = 0
28 neutral = 0
29 polarity = 0
30
31 for tweet in tweets:
32     print(tweet.text)
33     analysis = TextBlob(tweet.text)
34     polarity += analysis.sentiment.polarity
35
36     if(analysis.sentiment.polarity == 0):
37         neutral += 1
38
39     elif(analysis.sentiment.polarity < 0):
40         negative += 1
41
42     elif(analysis.sentiment.polarity > 0):
43         positive += 1
44
45
46 positive = percentage(positive, noOfSearchTerms)
47 neutral = percentage(neutral, noOfSearchTerms)
48 negative = percentage(negative, noOfSearchTerms)
49
50 positive = format(positive, '.2f')
51 neutral = format(neutral, '.2f')
52 negative = format(negative, '.2f')
53
54 print("positive:", positive, "negative:", negative, "neutral", neutral, "polarity:",
55     polarity)
56 print("how people are reacting on "+ searchTerm + " by analyzing " + str(
57     noOfSearchTerms) + "Tweets.")
58 labels = ['positive['+str(positive)+'%]', 'neutral['+str(neutral)+'%]', '
```



```

    negative['+str(negative)+'%']
59 sizes = [positive, neutral, negative]
60 colors = ['green', 'gold', 'red']
61 patches, text= plt.pie(sizes, colors=colors, startangle=90)
62 plt.legend(patches, labels, loc="best")
63 plt.title("how people are reacting on " + searchTerm + " by analyzing " + str(
    noOfSearchTerms) + " Tweets.")
64 plt.axis('equal')
65 plt.tight_layout()
66 plt.show()

```

6.3 Insertion of Cricbuzz Data to Database(Python Script)

batting_order	player	team	type_of_match	country
1	Virat Kohli	IND	batting_test	INDIA
2	Rohit Sharma	IND	batting_test	INDIA
3	Shane Watson	IND	batting_test	INDIA
4	Quinton de Kock	IND	batting_test	INDIA
5	Hashim Ahamd	IND	batting_test	INDIA
6	Wahab Raza	IND	batting_test	INDIA
7	Asif Ali Zafar	IND	batting_test	INDIA
8	Shoaib Akhtar	IND	batting_test	INDIA
9	Imran-ul-Faq	IND	batting_test	INDIA
10	Yasir Hameed	IND	batting_test	INDIA
11	Shoaib Akhtar	IND	batting_test	INDIA
12	Imran-ul-Faq	IND	batting_test	INDIA
13	Yasir Hameed	IND	batting_test	INDIA
14	Shoaib Akhtar	IND	batting_test	INDIA
15	Imran-ul-Faq	IND	batting_test	INDIA
16	Yasir Hameed	IND	batting_test	INDIA
17	Shoaib Akhtar	IND	batting_test	INDIA
18	Imran-ul-Faq	IND	batting_test	INDIA
19	Yasir Hameed	IND	batting_test	INDIA
20	Shoaib Akhtar	IND	batting_test	INDIA
21	Imran-ul-Faq	IND	batting_test	INDIA
22	Yasir Hameed	IND	batting_test	INDIA
23	Shoaib Akhtar	IND	batting_test	INDIA
24	Imran-ul-Faq	IND	batting_test	INDIA
25	Yasir Hameed	IND	batting_test	INDIA
26	Shoaib Akhtar	IND	batting_test	INDIA
27	Imran-ul-Faq	IND	batting_test	INDIA
28	Yasir Hameed	IND	batting_test	INDIA
29	Shoaib Akhtar	IND	batting_test	INDIA
30	Imran-ul-Faq	IND	batting_test	INDIA
31	Yasir Hameed	IND	batting_test	INDIA
32	Shoaib Akhtar	IND	batting_test	INDIA
33	Imran-ul-Faq	IND	batting_test	INDIA
34	Yasir Hameed	IND	batting_test	INDIA
35	Shoaib Akhtar	IND	batting_test	INDIA
36	Imran-ul-Faq	IND	batting_test	INDIA
37	Yasir Hameed	IND	batting_test	INDIA
38	Shoaib Akhtar	IND	batting_test	INDIA
39	Imran-ul-Faq	IND	batting_test	INDIA
40	Yasir Hameed	IND	batting_test	INDIA
41	Shoaib Akhtar	IND	batting_test	INDIA
42	Imran-ul-Faq	IND	batting_test	INDIA
43	Yasir Hameed	IND	batting_test	INDIA
44	Shoaib Akhtar	IND	batting_test	INDIA
45	Imran-ul-Faq	IND	batting_test	INDIA
46	Yasir Hameed	IND	batting_test	INDIA
47	Shoaib Akhtar	IND	batting_test	INDIA
48	Imran-ul-Faq	IND	batting_test	INDIA
49	Yasir Hameed	IND	batting_test	INDIA
50	Shoaib Akhtar	IND	batting_test	INDIA

Figure 6.4: Insertion of cricbuzz data in database

```

1
2 #THIS SCRIPT TAKES TIME TO COMMIT ON DATABASE(BATTING_TEST/ODO/T20 N OWLING
  TEST/ODI/T20)
3 import mysql.connector
4
5 #mysql connection
6 mydb = mysql.connector.connect(
7     host="localhost",
8     user="root",
9     passwd="",
10    database="prediction"
11 )
12

```

```
13 #reading batting n bowling of all type of match
14 batting_test = open("batting_test.csv",'r')
15 batting_odi = open("batting_odi.csv","r")
16 batting_t20 = open("batting_t20.csv","r")
17 bowling_test = open("bowling_test.csv","r")
18 bowling_odi = open("bowling_odi.csv","r")
19 bowling_t20 = open("bowling_t20.csv","r")
20
21 #for batting test
22 for x in batting_test:
23     #connection pointer
24     mycursor = mydb.cursor()
25     y = x.split(",")
26     sql = "INSERT INTO batting_test(position ,player ,rating ,type_of_match ,country )
27           VALUES(%s, %s, %s, %s, %s)"
28     val = (y[0],y[1],y[2],y[3],y[4])
29     mycursor.execute(sql , val)
30     mydb.commit()
31 #print("processing table:",x)
32
33 print("Batting test table insert")
34
35 #for batting odi
36 for x in batting_odi:
37     #connection pointer
38     mycursor = mydb.cursor()
39     y = x.split(",")
40     sql = "INSERT INTO batting_odi(position ,player ,rating ,type_of_match ,country )
41           VALUES(%s, %s, %s, %s, %s)"
42     val = (y[0],y[1],y[2],y[3],y[4])
43     mycursor.execute(sql , val)
44     mydb.commit()
45 #print("processing table:",x)
46
47 print("Batting odi table insert")
48
49 #for batting t20
50 for x in batting_t20:
51     #connection pointer
52     mycursor = mydb.cursor()
53     y = x.split(",")
54     sql = "INSERT INTO batting_t20(position ,player ,rating ,type_of_match ,country )
55           VALUES(%s, %s, %s, %s, %s)"
56     val = (y[0],y[1],y[2],y[3],y[4])
57     mycursor.execute(sql , val)
58     mydb.commit()
59
60 print("Batting t20 table insert")
61
62 #for bowling test
63 for x in bowling_test:
64     #connection pointer
65     mycursor = mydb.cursor()
66     y = x.split(",")
67     sql = "INSERT INTO bowling_test(position ,player ,rating ,type_of_match ,country )
68           VALUES(%s, %s, %s, %s, %s)"
69     val = (y[0],y[1],y[2],y[3],y[4])
70     mycursor.execute(sql , val)
71     mydb.commit()
72
73 print("Bowling test table insert")
```

```

70 #for bowling odi
71 for x in bowling_odi:
72 #connection pointer
73 mycursor = mydb.cursor()
74 y = x.split(",")
75 sql = "INSERT INTO bowling_odi(position ,player ,rating ,type_of_match ,country)
       VALUES(%s , %s , %s , %s , %s)"
76 val = (y[0],y[1],y[2],y[3],y[4])
77 mycursor.execute(sql , val)
78 mydb.commit()
79 print("Bowling odi table insert")
80
81 #for bowling t20
82 for x in bowling_t20:
83 #connection pointer
84 mycursor = mydb.cursor()
85 y = x.split(",")
86 sql = "INSERT INTO bowling_t20(position ,player ,rating ,type_of_match ,country)
       VALUES(%s , %s , %s , %s , %s)"
87 val = (y[0],y[1],y[2],y[3],y[4])
88 mycursor.execute(sql , val)
89 mydb.commit()
90 print("Bowling t20 table insert")

```

6.4 Cricbuzz Different Data Merge Into One Super Data

player_id	player_name	rating	type_of_match	country
1	Adam Frost	308	AUSTRALIA	
2	Adam Frost	600	AUSTRALIA	
3	Adam Frost	302	AUSTRALIA	
4	Adam Frost	342	SOUTH AFRICA	
5	Adam Frost	129	BANGLADESH	
6	Adam Frost	600	SRI LANKA	
7	Adam Frost	400	NEW ZEALAND	
8	Adam Frost	400	AUSTRALIA	
9	Adam Frost	400	AUSTRALIA	
10	Adam Frost	400	INDIA	
11	Adam Frost	400	INDIA	
12	Adam Frost	300	INDIA	
13	Adam Frost	400	INDIA	
14	Adam Frost	400	INDIA	
15	Adam Frost	400	INDIA	
16	Adam Frost	400	INDIA	
17	Adam Frost	400	INDIA	
18	Adam Frost	400	INDIA	
19	Adam Frost	400	INDIA	
20	Adam Frost	400	INDIA	
21	Adam Frost	400	INDIA	
22	Adam Frost	400	INDIA	
23	Adam Frost	400	INDIA	
24	Adam Frost	400	INDIA	
25	Adam Frost	400	INDIA	
26	Adam Frost	400	INDIA	
27	Adam Frost	400	INDIA	
28	Adam Frost	400	INDIA	
29	Adam Frost	400	INDIA	
30	Adam Frost	400	INDIA	
31	Adam Frost	400	INDIA	
32	Adam Frost	400	INDIA	

Figure 6.5: All-in-One Data For Selection

```
1 import mysql.connector
2
3 #mysql connection
4 mydb = mysql.connector.connect(
5     host="localhost",
6     user="root",
7     passwd="",
8     database="prediction"
9 )
10
11 all_rank = []
12
13 mycursor = mydb.cursor()
14 #batting_test
15 mycursor.execute("SELECT player ,rating ,country FROM batting_test")
16 bat = mycursor.fetchall()
17 #batting odi
18 mycursor.execute("SELECT player ,rating ,country FROM batting_odi")
19 bao = mycursor.fetchall()
20 #batting_t20
21 mycursor.execute("SELECT player ,rating ,country FROM batting_t20")
22 bat20 = mycursor.fetchall()
23 #bowling_test
24 mycursor.execute("SELECT player ,rating ,country FROM bowling_test")
25 bot = mycursor.fetchall()
26 #bowling odi
27 mycursor.execute("SELECT plaYer ,rating ,country FROM bowling_odi")
28 boo = mycursor.fetchall()
29 #bowling t20
30 mycursor.execute("SELECT player ,rating ,country FROM bowling_t20")
31 bot20 = mycursor.fetchall()
32
33 for t in bat:
34     all_rank.append(t)
35
36 for u in bao:
37     all_rank.append(u)
38
39 for v in bat20:
40     all_rank.append(v)
41
42 for w in bot:
43     all_rank.append(w)
44
45 for x in boo:
46     all_rank.append(x)
47
48 for y in bot20:
49     all_rank.append(y)
50
51 all_rank.sort()
52
53 for x in range(len(all_rank)):
54     sql = "INSERT INTO test(player_id ,player ,rating ,country) VALUES(%s , %s , %s , %s)
55         "
56     val = (x, all_rank[x][0], all_rank[x][1], all_rank[x][2])
57     mycursor.execute(sql, val)
58     print("q:", x)
59 mydb.commit()
```

6.5 Removing Tuple Duplication- Final Rank (Averaging all Data)

player_id	player	rating	country
1	Aaron Phangiso	503	SOUTH AFRICA
4	Abdul Razaq	129	PAKISTAN
5	Ali Javid	188	PAKISTAN
6	Adam Baker	492	NEW ZEALAND
7	Adam Taylor	505	AUSTRALIA
8	Adrian Smith	388	ENGLAND
12	Adip, Ivan	381	AFGHANISTAN
14	Abdul Razaq	478	PAKISTAN
15	Adrian Smith	505	ENGLAND
16	Adam Baker	388	NEW ZEALAND
18	Adam Taylor	381	ENGLAND
19	Adam Smith	381	ENGLAND
20	Adam Smith	381	ENGLAND
21	Adam Smith	381	ENGLAND
22	Adam Smith	381	ENGLAND
23	Adam Smith	381	ENGLAND
24	Adam Smith	381	ENGLAND
25	Adam Smith	381	ENGLAND
26	Adam Smith	381	ENGLAND
27	Adam Smith	381	ENGLAND
28	Adam Smith	381	ENGLAND
29	Adam Smith	381	ENGLAND
30	Adam Smith	381	ENGLAND
31	Adam Smith	381	ENGLAND
32	Adam Smith	381	ENGLAND
33	Adam Smith	381	ENGLAND
34	Adam Smith	381	ENGLAND
35	Adam Smith	381	ENGLAND
36	Adam Smith	381	ENGLAND
37	Adam Smith	381	ENGLAND

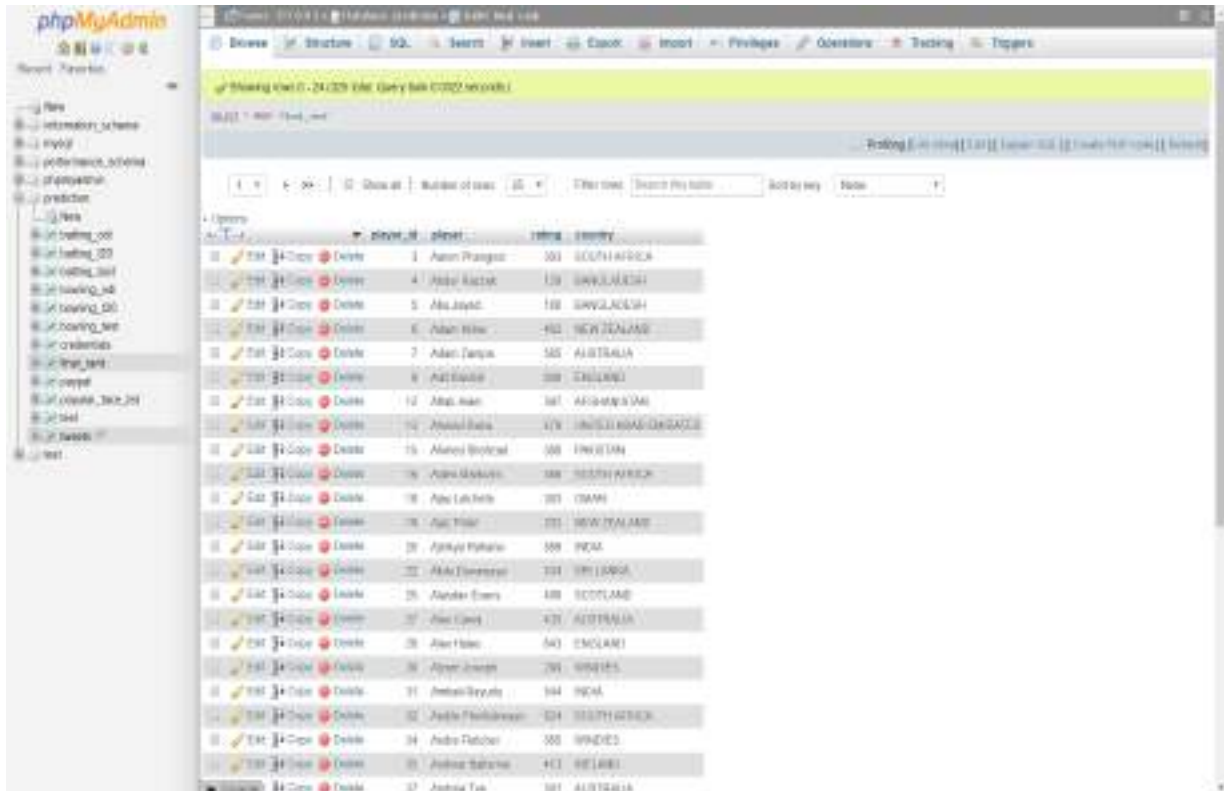
Figure 6.6: Final Rank Table

```

1  import mysql.connector
2  import csv
3
4  #mysql connection
5  mydb = mysql.connector.connect(
6      host="localhost",
7      user="root",
8      passwd="",
9      database="prediction"
10 )
11
12 mycursor = mydb.cursor()
13
14 mycursor.execute("SELECT player_id ,player ,AVG(rating) ,country FROM test GROUP BY
15     player")
16 final_all_rank = mycursor.fetchall();
17
18 #final rank upload
19 for x in range(len(final_all_rank)):
20     sql = "INSERT INTO final_rank(player_id ,player ,rating ,country) VALUES(%s , %s , %s , %s)"
21     val = (final_all_rank[x][0], final_all_rank[x][1], final_all_rank[x][2],
22           final_all_rank[x][3])
23     mycursor.execute(sql, val)
24     print("FINAL RANK UPLOADED")
25
26 mydb.commit()

```

6.6 Final Rank Player Details to Tweets Table



player_id	player	rating	country
1	Aaron Phangiso	503	SOUTH AFRICA
4	Abdul Razaq	139	PAKISTAN
5	Abu Jayed	139	BAHRAIN
6	Adam Smith	492	NEW ZEALAND
7	Adam Taylor	545	AUSTRALIA
8	Adrian Brown	388	ENGLAND
12	Adnan Khan	387	AFGHANISTAN
13	Adnan Khan	476	INDONESIA
15	Adnan Khan	500	INDONESIA
16	Adnan Khan	500	INDONESIA
18	Adnan Khan	500	INDONESIA
19	Adnan Khan	500	INDONESIA
20	Adnan Khan	500	INDONESIA
21	Adnan Khan	500	INDONESIA
22	Adnan Khan	500	INDONESIA
23	Adnan Khan	500	INDONESIA
24	Adnan Khan	500	INDONESIA
25	Adnan Khan	500	INDONESIA
26	Adnan Khan	500	INDONESIA
27	Adnan Khan	500	INDONESIA
28	Adnan Khan	500	INDONESIA
29	Adnan Khan	500	INDONESIA
30	Adnan Khan	500	INDONESIA
31	Adnan Khan	500	INDONESIA
32	Adnan Khan	500	INDONESIA
33	Adnan Khan	500	INDONESIA
34	Adnan Khan	500	INDONESIA
35	Adnan Khan	500	INDONESIA
36	Adnan Khan	500	INDONESIA
37	Adnan Khan	500	INDONESIA

Figure 6.7: Combining rank table and tweets table

```

1  import mysql.connector
2  import csv
3
4  #mysql connection
5  mydb = mysql.connector.connect(
6      host="localhost",
7      user="root",
8      passwd="",
9      database="prediction"
10 )
11
12 mycursor = mydb.cursor()
13
14 mycursor.execute("SELECT * FROM final_rank ORDER BY rating DESC")
15 #final_all_rank_row
16 farr = mycursor.fetchall()
17
18 #tweets feeder
19 for x in range(len(farr)):
20     sql = "INSERT INTO tweets(player_id,player,rating,country) VALUES(%s, %s, %s, %s)"
21     val = (farr[x][0],farr[x][1],farr[x][2],farr[x][3])
22     mycursor.execute(sql, val)
23
24 mydb.commit()
25
26 #generating final all rank csv which can be used for another table
27 with open('final_all_rank','w',newline='') as csv_final:

```

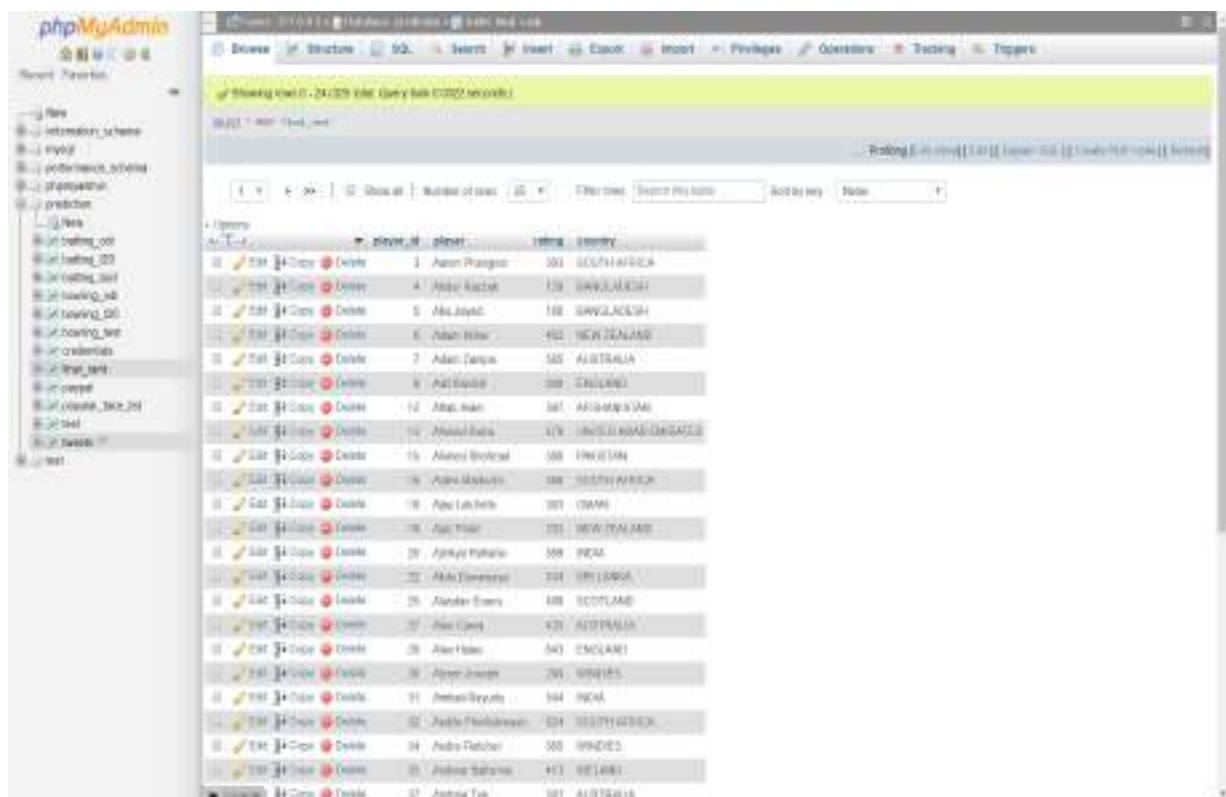


```

28 fieldnames = [ 'Player_id', 'Player', 'Rating', 'Country' ]
29 writer = csv.DictWriter(csv_final, fieldnames=fieldnames)
30 writer.writeheader()
31 for i in range(len(farr)):
32     writer.writerow({ 'Player_id': farr[i][0], 'Player': farr[i][1], 'Rating': farr[i]
33                       ][2], 'Country': farr[i][3], })
34 csv_final.close();

```

6.7 Updating Positive Sentiment and Overall Sentiment Type Into Tweets Table by Using Tweepy Script



tweet_id	player_id	player	rating	country
1	1	Aaron Phangiso	503	SOUTH AFRICA
2	4	Abhinav Rastogi	129	INDIA
3	5	Abhinav Rastogi	129	INDIA
4	6	Abhinav Rastogi	129	INDIA
5	7	Abhinav Rastogi	129	INDIA
6	8	Abhinav Rastogi	129	INDIA
7	9	Abhinav Rastogi	129	INDIA
8	10	Abhinav Rastogi	129	INDIA
9	11	Abhinav Rastogi	129	INDIA
10	12	Abhinav Rastogi	129	INDIA
11	13	Abhinav Rastogi	129	INDIA
12	14	Abhinav Rastogi	129	INDIA
13	15	Abhinav Rastogi	129	INDIA
14	16	Abhinav Rastogi	129	INDIA
15	17	Abhinav Rastogi	129	INDIA
16	18	Abhinav Rastogi	129	INDIA
17	19	Abhinav Rastogi	129	INDIA
18	20	Abhinav Rastogi	129	INDIA
19	21	Abhinav Rastogi	129	INDIA
20	22	Abhinav Rastogi	129	INDIA
21	23	Abhinav Rastogi	129	INDIA
22	24	Abhinav Rastogi	129	INDIA
23	25	Abhinav Rastogi	129	INDIA
24	26	Abhinav Rastogi	129	INDIA
25	27	Abhinav Rastogi	129	INDIA
26	28	Abhinav Rastogi	129	INDIA
27	29	Abhinav Rastogi	129	INDIA
28	30	Abhinav Rastogi	129	INDIA
29	31	Abhinav Rastogi	129	INDIA
30	32	Abhinav Rastogi	129	INDIA
31	33	Abhinav Rastogi	129	INDIA
32	34	Abhinav Rastogi	129	INDIA
33	35	Abhinav Rastogi	129	INDIA
34	36	Abhinav Rastogi	129	INDIA
35	37	Abhinav Rastogi	129	INDIA

Figure 6.8: Updation of table

```

1 from textblob import TextBlob
2 import sys, tweepy
3 import matplotlib.pyplot as plt
4 import mysql.connector
5
6 #db connection
7 mydb = mysql.connector.connect(
8     host="localhost",
9     user="root",
10    passwd="",
11    database="prediction"
12 )
13
14 mycursor = mydb.cursor()

```

```

15
16 mycursor.execute("SELECT player FROM tweets ORDER BY rating DESC")
17 #top 50 player with ref rating
18 tpr = mycursor.fetchall()
19
20 #Making list of top 50 player
21 list_of_tpr = []
22 for x in range(50):
23     list_of_tpr.append(tpr[x])
24
25 def percentage(part, whole):
26     return 100*float(part)/float(whole)
27
28 #api consumer and access key
29 consumerKey="ZeGyqaUBzYExcO4MHClfBaMI"
30 consumerSecret="edb3zFTHtjOLNzuaR3HUIsTyBR05M01UMaxkqJhuGQsF1fVn55"
31 accessToken="1098235662753398784-2177Cg9fViEcJduSVTxEaxeoKdGZYv"
32 accessTokenSecret="bqXBx4LiiviNy5euaNel8nFPlbrbQwottD8DXcBtKQ8Cx"
33
34 #Authentication
35 auth = tweepy.OAuthHandler(consumerKey, consumerSecret)
36 auth.set_access_token(accessToken, accessTokenSecret)
37 #add wait list
38 api = tweepy.API(auth, wait_on_rate_limit=True)
39
40 for st in range(len(list_of_tpr)):
41     #searchTerm = input("enter keyword/hashtag to search: ")
42     searchTerm = st
43     #noOfSearchTerms = int(input("how many tweets to analyze: "))
44     noOfSearchTerms = 50
45     tweets = tweepy.Cursor(api.search, q=searchTerm, lang="en").items(
46         noOfSearchTerms)
47     #sentiment count
48     positive = 0
49     negative = 0
50     neutral = 0
51     polarity = 0
52
53     for tweet in tweets:
54         print(tweet.text)
55         analysis = TextBlob(tweet.text)
56         polarity += analysis.sentiment.polarity
57
58         if(analysis.sentiment.polarity == 0):
59             neutral += 1
60
61         elif(analysis.sentiment.polarity < 0):
62             negative += 1
63
64         elif(analysis.sentiment.polarity > 0):
65             positive += 1
66
67     positive = percentage(positive, noOfSearchTerms)
68     neutral = percentage(neutral, noOfSearchTerms)
69     negative = percentage(negative, noOfSearchTerms)
70
71     positive = format(positive, '.2f')
72     neutral = format(neutral, '.2f')
73     negative = format(negative, '.2f')
74     ps = ""

```



```
74 print("positive:", positive, "negative:", negative, "neutral", neutral, "polarity:",  
75      polarity)  
76 if(neutral > positive and neutral > negative):  
77     ps = "Neutral"  
78 elif(negative > positive and negative > neutral):  
79     ps = "negative"  
80 elif(positive > negative and positive > neutral):  
81     ps = "positive"  
82  
83     print("_____")  
84     print(ps)  
85     positive = int(float(positive))  
86     sql_tpr = "UPDATE tweets SET positive_sentiment = %s ,overall_sentiment = %  
87               s WHERE player = %s"  
88     val_sql_tpr = (positive, ps, list_of_tpr[st][0])  
89     mycursor.execute(sql_tpr, val_sql_tpr)  
90     mydb.commit()  
91     print("_____")  
92 print("_____")
```

Chapter 7

System Testing

First system will check the parsing function if that is implemented successfully so it will go the fetch function and take the data from the websites based on the structure that we have mentioned in the fetch function. If the fetch function implemented successfully then it will go to the store function and store the data into the database.

7.1 Test Cases and Test Results

Test ID	Test Case Title	Test Condition	System Behavior	Expected Result
T01	Testing Library	Is it working?	Loaded websites after fetching	Successfully
T02	Test function1	Parsing websites	Completely scraped the data	Successfully
T03	Test function2	Store Data	Stored into the database	Successfully

Table 7.1: Table for Test Cases and Results

7.2 Test Case

Title: Scraping the data from the websites successfully.

Description: Before implementation part testing are also important for any system so before implementation of the project first we have to also test the cases that we are going to implement in our project. Our project is first scrap the data from websites and stored into the database so to the website's to scrap we have used Beautiful soup. Once we will integrate this library only parsing part will be remaining that we will get from the website's structure such as Html tag which is used in website's to built that is about to scrap. Once our fetching part

will be done then we have to check that we are getting the data from the website's which we have targeted based on the web-site's tag such as html tag.

Here for testing purpose we have targeted a website's that is related to the product advertisement. When we are implementing the testing part so we are successfully getting the data we require.

Precondition: There is no authentication are required to the users.

Assumption: A supported browser is being used.

Test Steps:

- I Implementing the fetching function.
- II Getting the data from the websites.
- III Implementing the store function
- IV Stored the data into the database.

Expected Result: To get the data from the websites based on the websites structure.

Actual Result: We are successfully getting the data from the websites that we have targeted to scrap the data such as tweets and rank according to the players name.

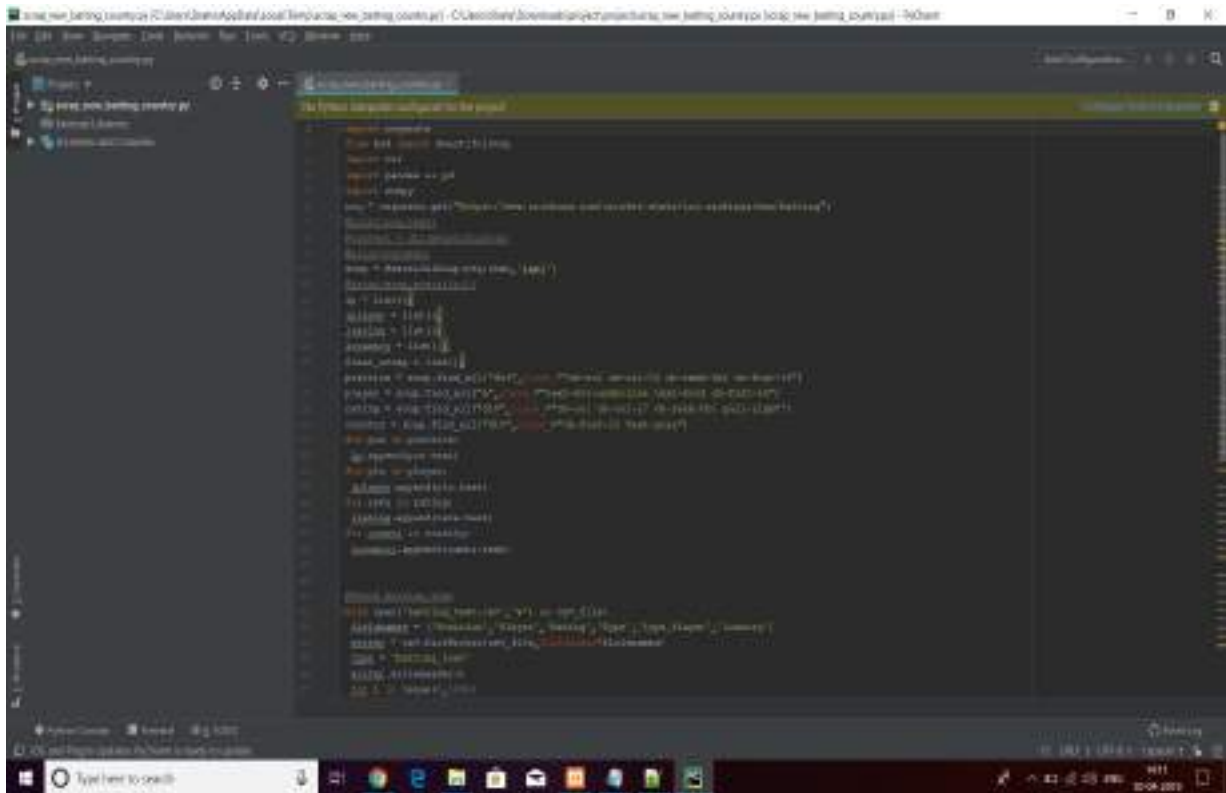


Figure 7.1: Successful loading of website

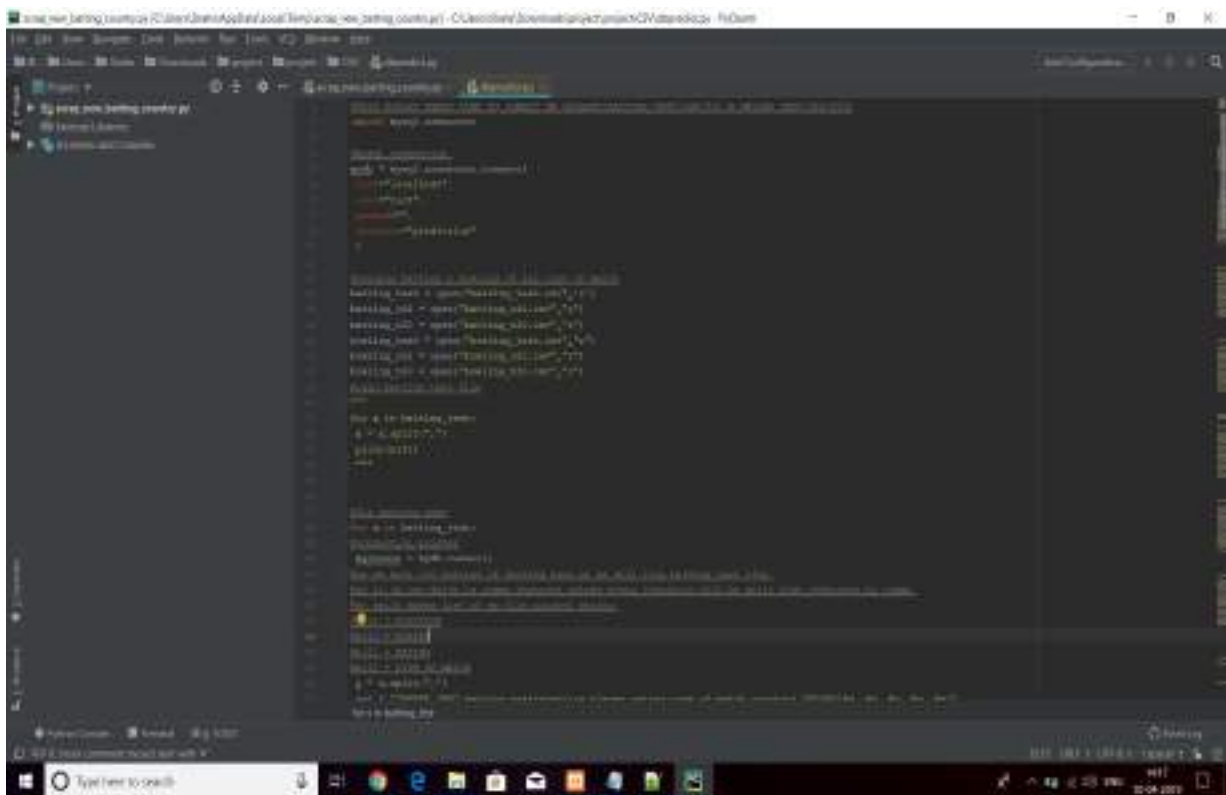
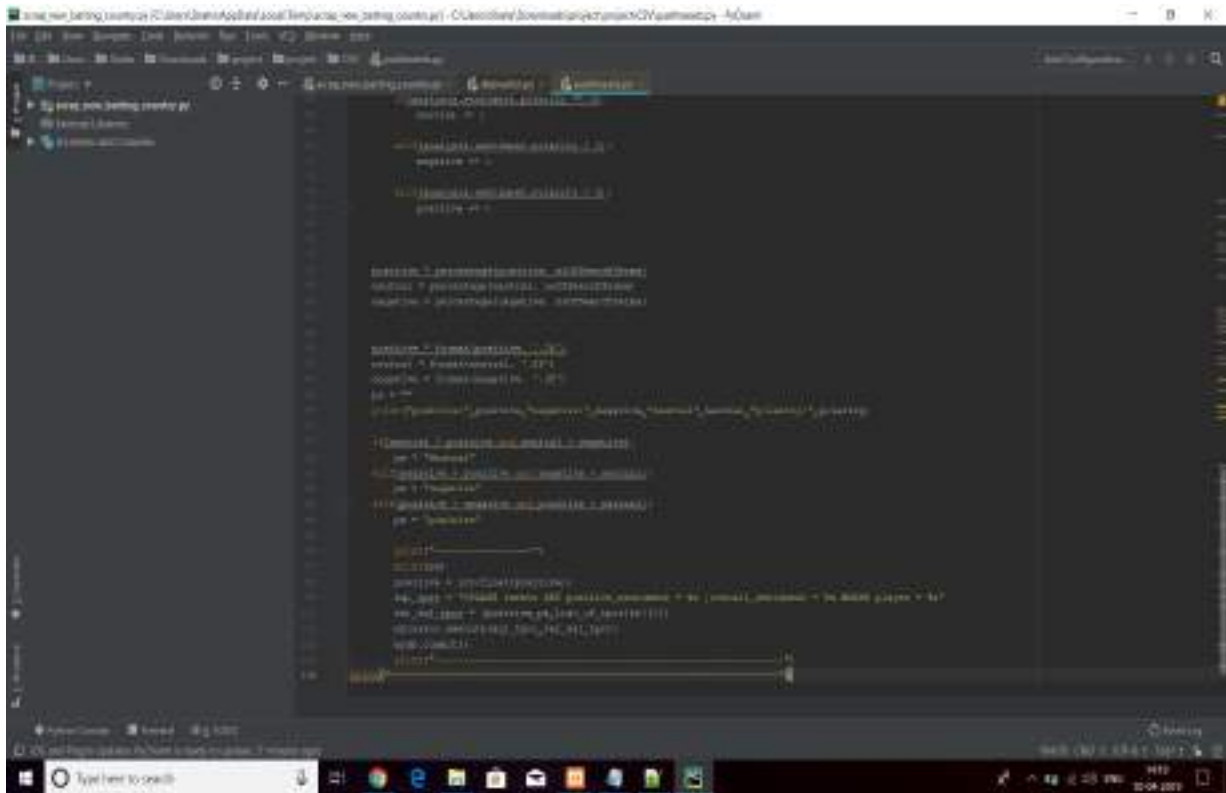


Figure 7.2: Successful scraping of the data



The screenshot shows a Jupyter Notebook interface with a dark theme. The left sidebar displays a file explorer with a folder named 'data' containing a file 'data.csv'. The main area shows a code cell with the following Python code:

```
import pandas as pd
import numpy as np
import os

# Load the data
data = pd.read_csv('data.csv')

# Display the first few rows
data.head()

# Check the data types
data.dtypes

# Save the data to a new file
data.to_csv('data_new.csv', index=False)
```

The output of the code cell shows the first few rows of the data, the data types, and a confirmation message that the data has been saved to 'data_new.csv'.

Figure 7.3: Successful data storage

Chapter 8

Screenshots of Project

8.1 Front End of the System

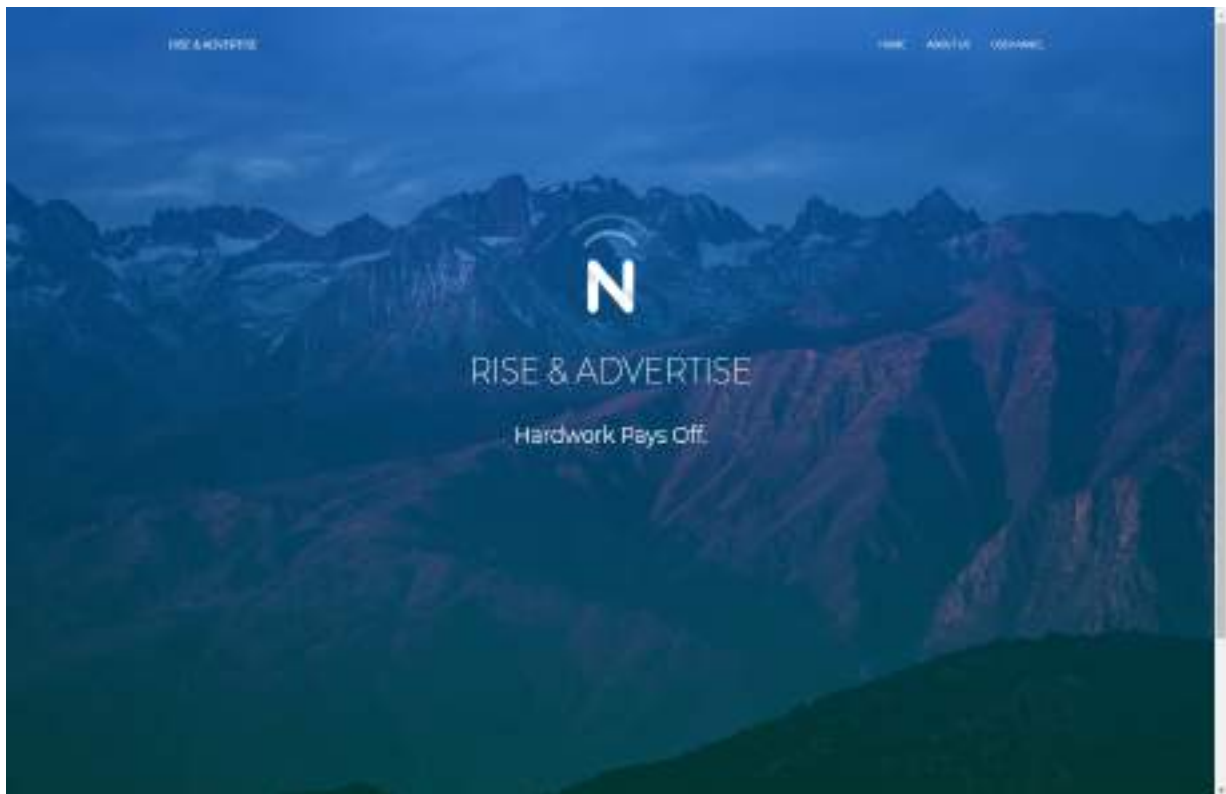


Figure 8.1: Home Page

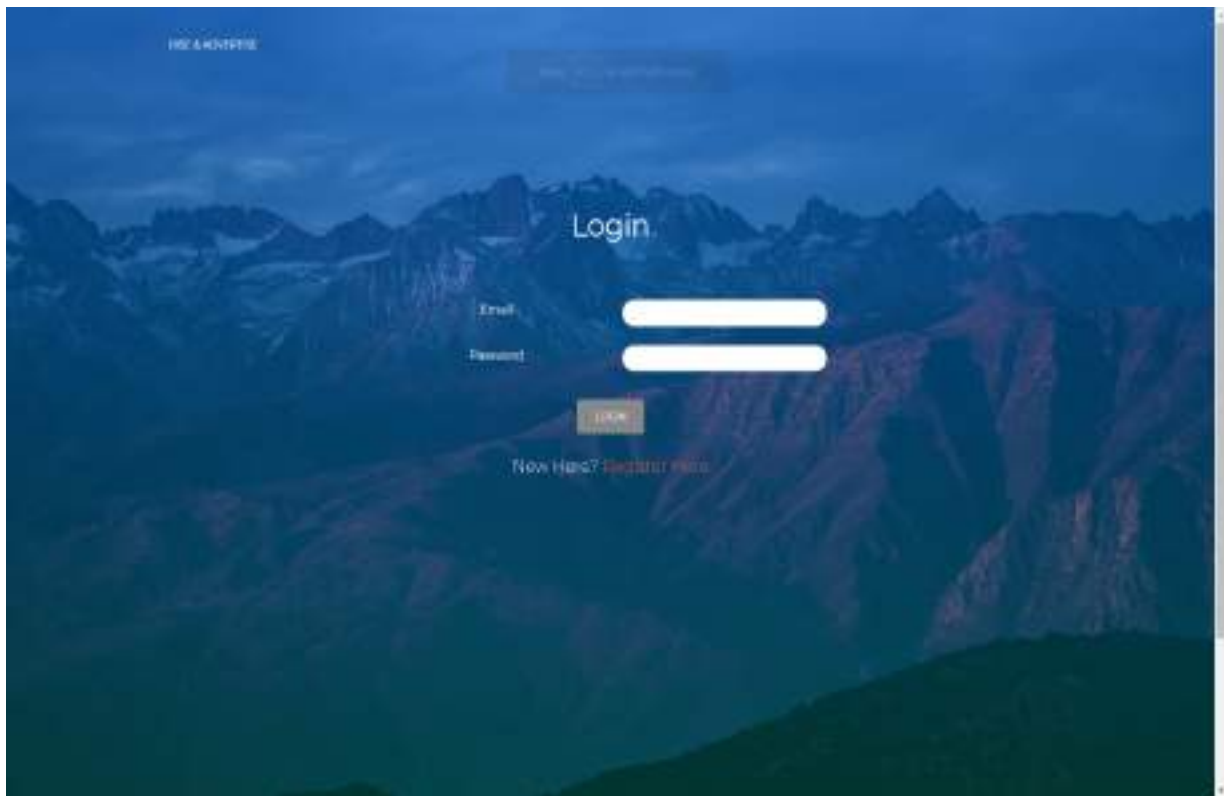


Figure 8.2: Login Page

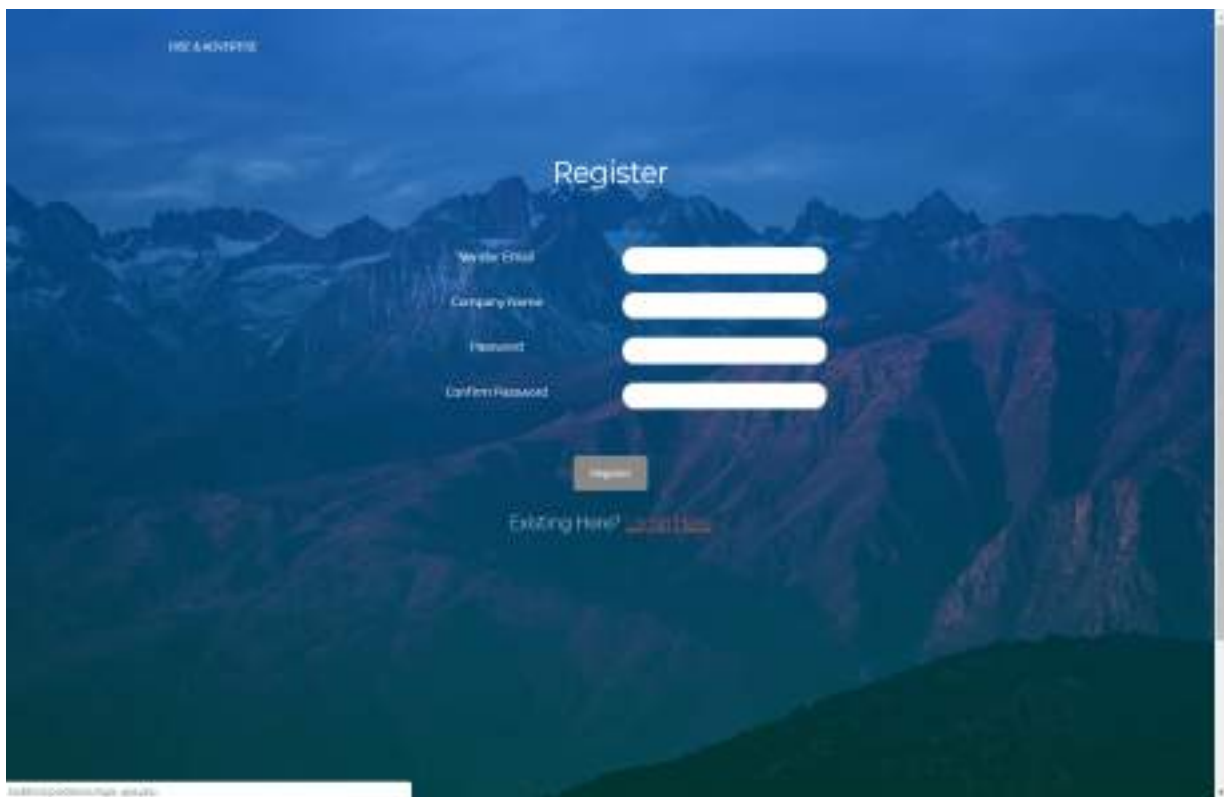


Figure 8.3: Registration Page

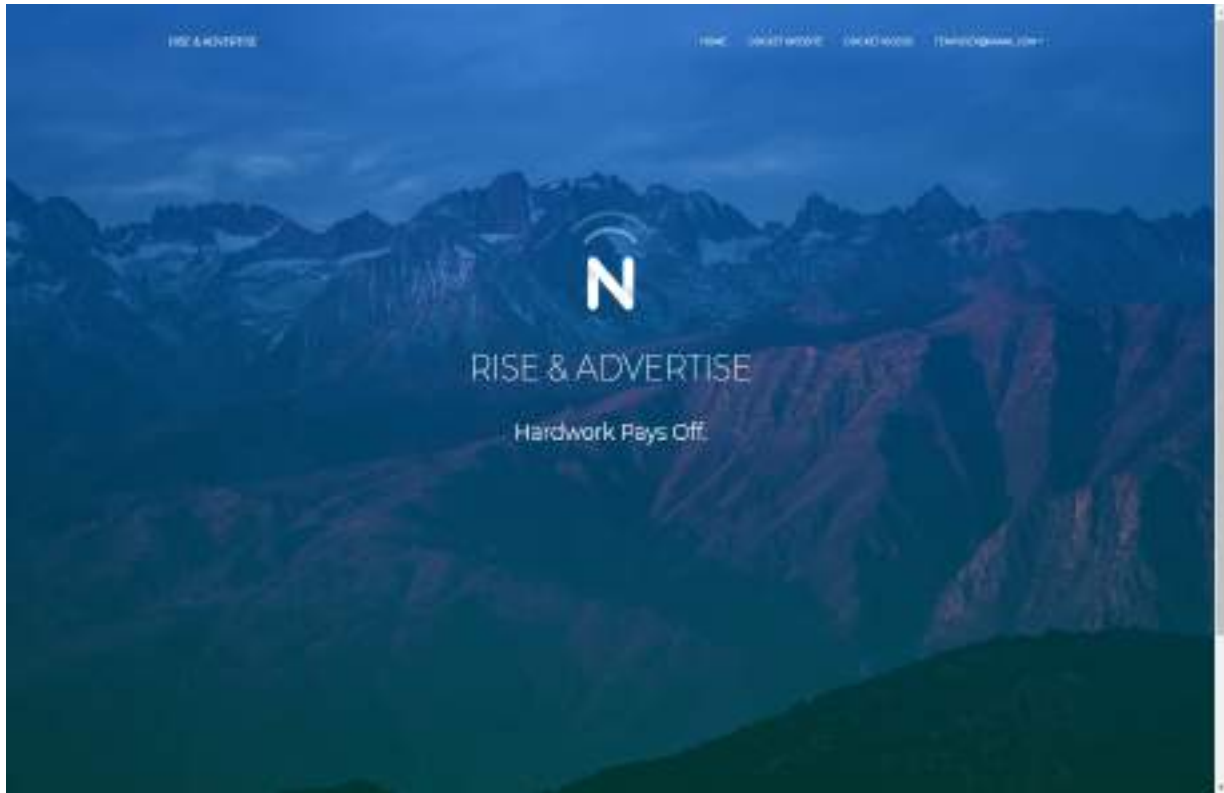


Figure 8.4: Paid user home page

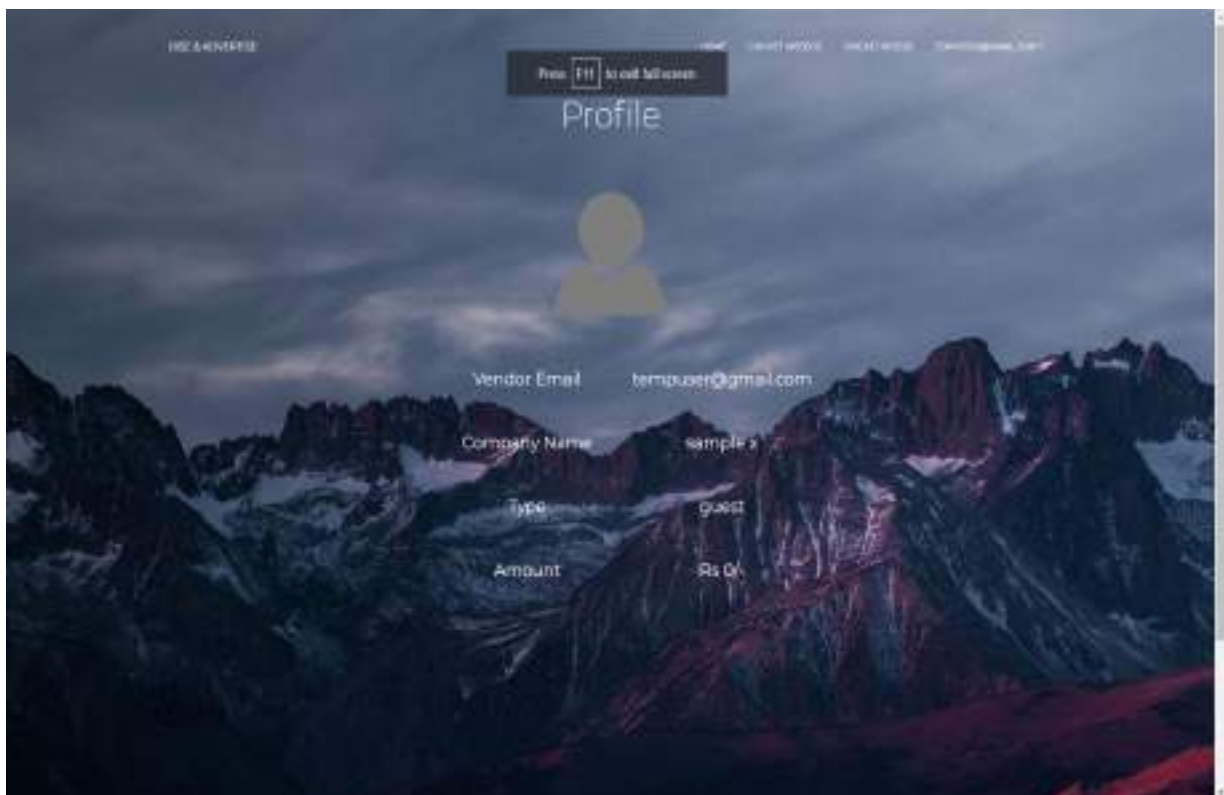
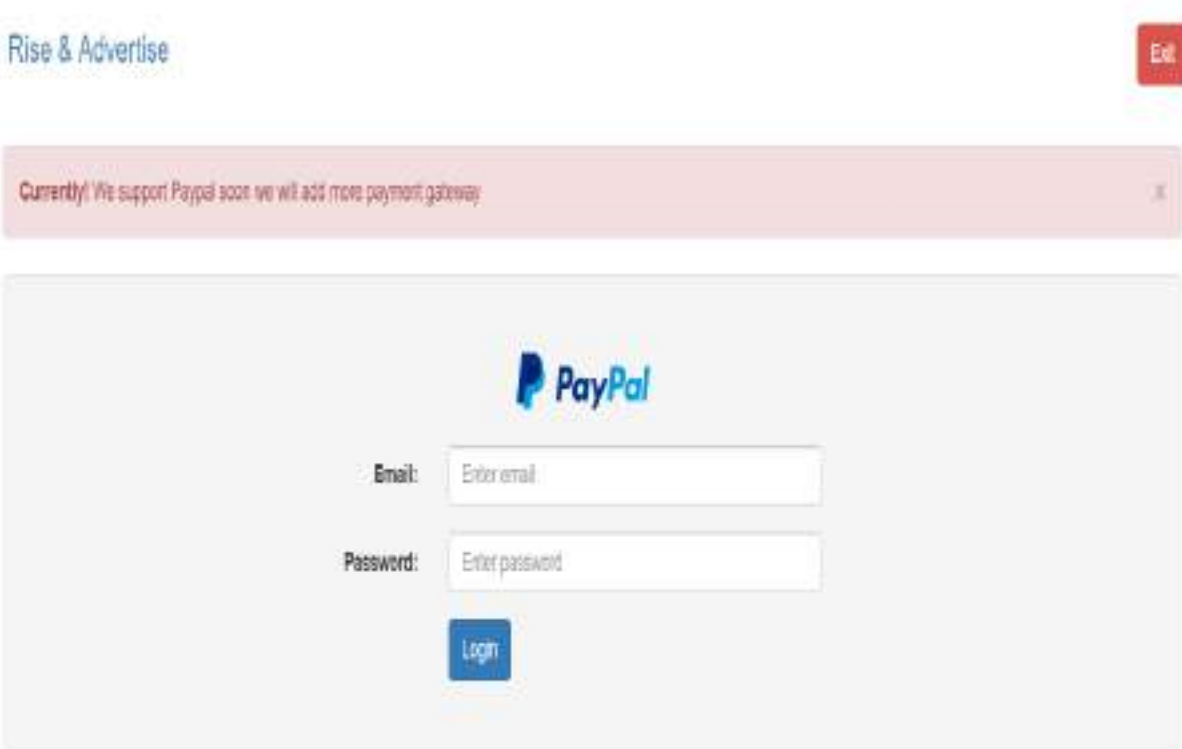


Figure 8.5: Profile page



Rise & Advertise

Currently! We support Paypal soon we will add more payment gateway

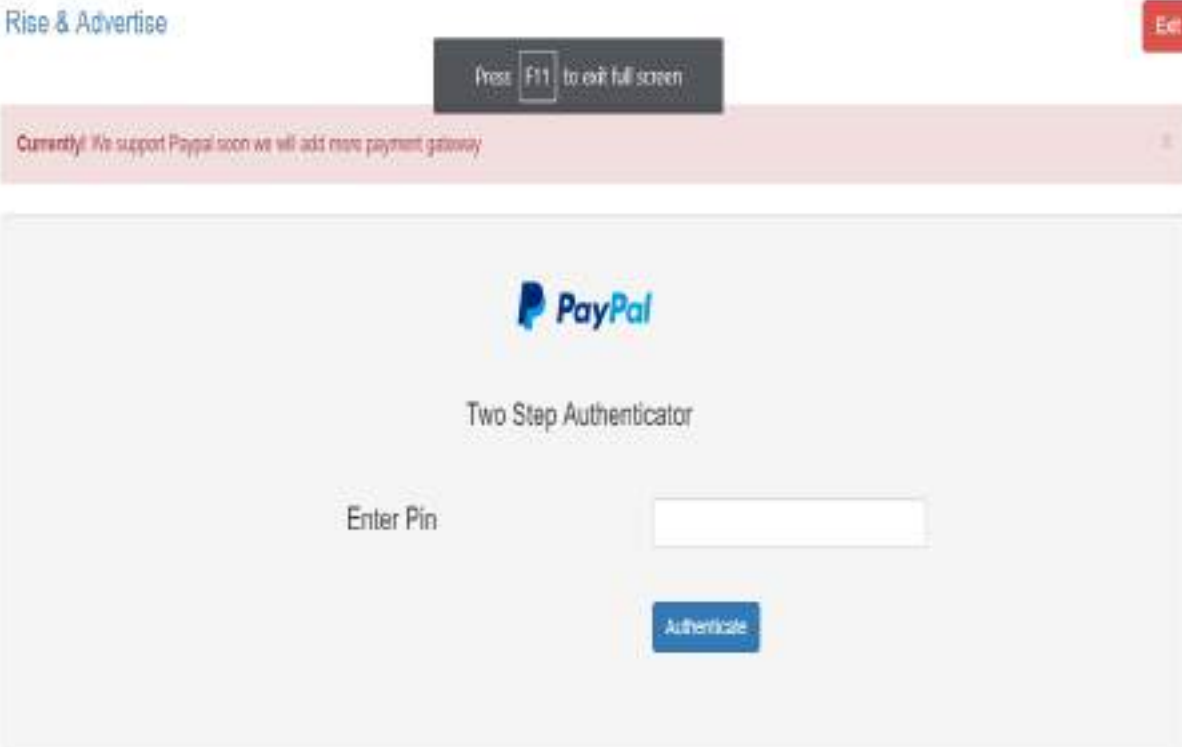
PayPal

Email:

Password:

Login

Figure 8.6: Payment gateway page



Rise & Advertise

Press F11 to exit full screen

Currently! We support Paypal soon we will add more payment gateway

PayPal

Two Step Authenticator

Enter Pin

Authenticate

Figure 8.7: Payment gateway authenticator page

Rise & Advertise Exit

Currently! We support Paypal soon we will add more payment gateway

PayPal

Hello,usertemp@gmail.com Amount: 10,000 INR

Subscription of Rise & Advertise

1 month @ 500/-

Process

Figure 8.8: Subscription page

Rise & Advertise Exit

Currently! We support Paypal soon we will add more payment gateway

PayPal

Hello,usertemp@gmail.com Amount: 10,000 INR

Subscription of Rise & Advertise

Success

Are you sure to pay for 1 Month @ 500/-

Pay Us Cancel

Figure 8.9: Pay us page

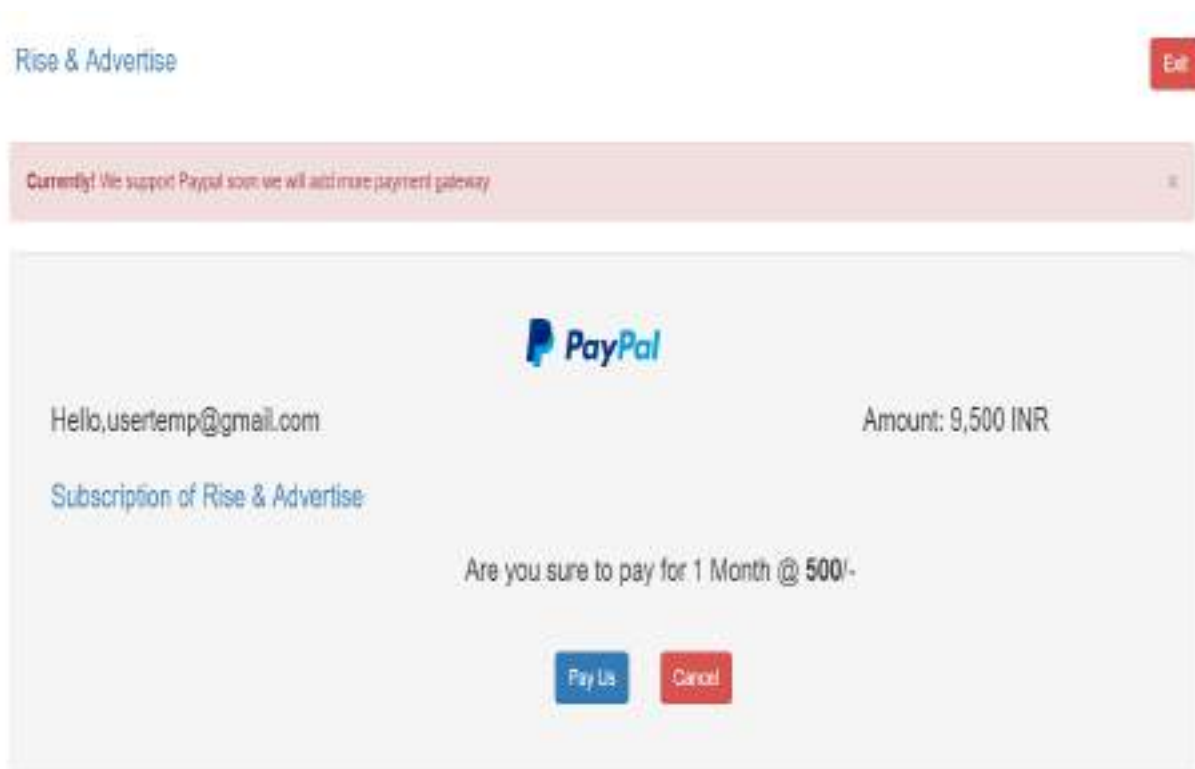


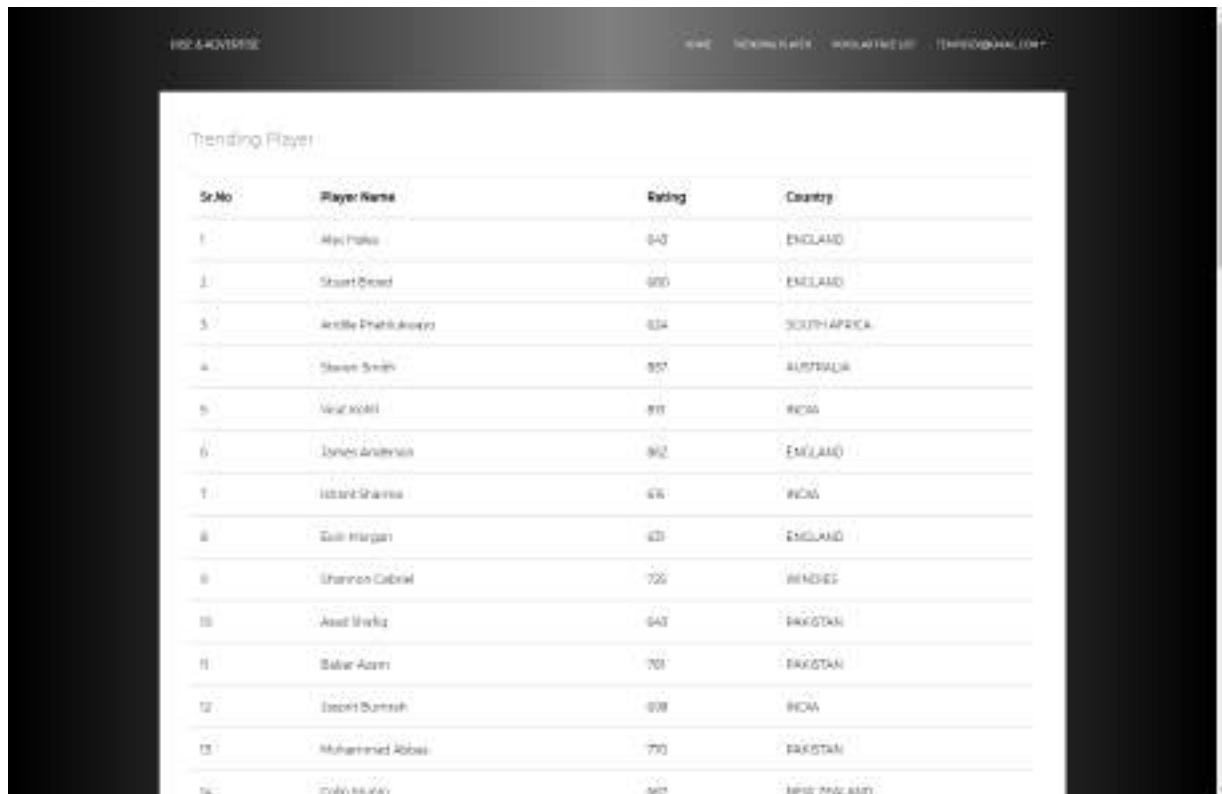
Figure 8.10: Pay us confirmation

8.2 Database Storage

id	username	password	email	first_name	last_name	gender	country
1	Glennox Gabriel	115	343	811	Shangyu Soft Watch Dream Facewatch Perfume	8000000	INDIA
2	Dimitri Kuznetsov	004	000	580	Chips Facewatch Chocolate Model	8000000	INDIA
3	Machin Kishan	822	40	311	Green Facewatch Perfume Watch Car Bags	8000000	INDIA
4	Yashu Sanyal	947	450	585	Black Car Watch Car Bags	8000000	INDIA
5	Yashu Sanyal	947	000	821	Chips Green Facewatch Sports Shoes	8000000	INDIA
6	Yashu Sanyal	947	000	821	Black Car Watch Car Bags	8000000	INDIA
7	Glennox Gabriel	115	810	840	Green Facewatch Perfume Soft Watch Sports Shoes	8000000	INDIA
8	Fatima Zahra	107	800	540	Green Facewatch Perfume Sports Shoes	8000000	INDIA
9	Helen Ann	811	110	80	Watch Bags	8000000	INDIA
10	Helen Ann	811	810	820	Black Car Watch Car Bags	8000000	INDIA
11	Yashu Sanyal	947	110	700	Watch Sports Green Facewatch Perfume	8000000	INDIA
12	Yashu Sanyal	947	810	810	Green Facewatch Perfume Soft Watch Sports Shoes	8000000	INDIA
13	Yashu Sanyal	947	000	821	Green Facewatch Perfume Sports Shoes	8000000	INDIA
14	Yashu Sanyal	947	000	821	Green Facewatch Perfume Sports Shoes	8000000	INDIA
15	Yashu Sanyal	947	000	821	Green Facewatch Perfume Sports Shoes	8000000	INDIA
16	Yashu Sanyal	947	000	821	Green Facewatch Perfume Sports Shoes	8000000	INDIA
17	Yashu Sanyal	947	000	821	Green Facewatch Perfume Sports Shoes	8000000	INDIA
18	Yashu Sanyal	947	000	821	Green Facewatch Perfume Sports Shoes	8000000	INDIA
19	Yashu Sanyal	947	000	821	Green Facewatch Perfume Sports Shoes	8000000	INDIA
20	Yashu Sanyal	947	000	821	Green Facewatch Perfume Sports Shoes	8000000	INDIA
21	Yashu Sanyal	947	000	821	Green Facewatch Perfume Sports Shoes	8000000	INDIA
22	Yashu Sanyal	947	000	821	Green Facewatch Perfume Sports Shoes	8000000	INDIA
23	Yashu Sanyal	947	000	821	Green Facewatch Perfume Sports Shoes	8000000	INDIA
24	Yashu Sanyal	947	000	821	Green Facewatch Perfume Sports Shoes	8000000	INDIA
25	Yashu Sanyal	947	000	821	Green Facewatch Perfume Sports Shoes	8000000	INDIA

Figure 8.11: Stored database

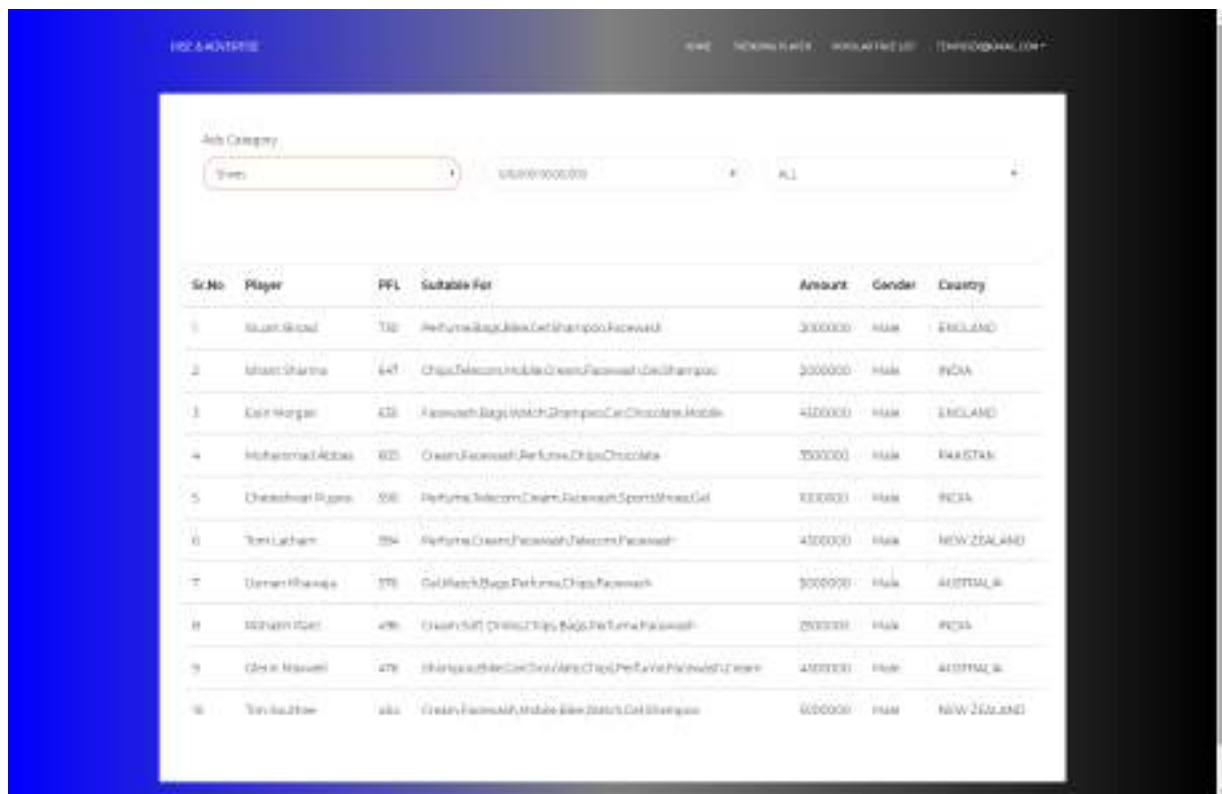
8.3 Trending Player and Popular Facelist



Web application interface showing a 'Trending Player' facelist. The table displays the following data:

Sr.No	Player Name	Rating	Country
1	Ally Pollock	843	ENGLAND
2	Stuart Broad	885	ENGLAND
3	Andile Mkhuzeni	834	SOUTH AFRICA
4	Shane Smith	857	AUSTRALIA
5	Virat Kohli	898	INDIA
6	Dinesh Karthikeyan	862	ENGLAND
7	Ishan Sharma	688	INDIA
8	Eoin Morgan	628	ENGLAND
9	Shannon Gabriel	725	IRELAND
10	Azeez Sheikh	643	PAKISTAN
11	Salim Akram	701	PAKISTAN
12	Imran Tahir	698	INDIA
13	Muhammad Akmal	790	PAKISTAN
14	Colin Munro	697	NEW ZEALAND

Figure 8.12: Trending players facelist



Web application interface showing a 'Vendors filtered popular facelist'. The table displays the following data:

Sr.No	Player	PFL	Suitable For	Amount	Gender	Country
1	Stuart Broad	732	Perfume, Soap, Socks, Gel, Shampoo, Facewash	300000	Male	ENGLAND
2	Ishan Sharma	647	Chips, Telecom, Mobile, Cream, Facewash, Gel, Shampoo	300000	Male	INDIA
3	Eoin Morgan	628	Facewash, Soap, Watch, Shampoo, Car, Chocolate, Mobile	410000	Male	ENGLAND
4	Muhammad Akmal	625	Cream, Facewash, Perfume, Chips, Chocolate	350000	Male	PAKISTAN
5	Chandrasekar Pillay	598	Perfume, Telecom, Cream, Facewash, Sports Shoes, Gel	300000	Male	INDIA
6	Tom Latham	584	Perfume, Cream, Facewash, Telecom, Facewash	450000	Male	NEW ZEALAND
7	Darren Lehmann	575	Gel, Watch, Soap, Perfume, Chips, Facewash	500000	Male	AUSTRALIA
8	Shane Watson	496	Cream, Gel, Cream, Soap, Soap, Perfume, Facewash	250000	Male	INDIA
9	Chris Insole	479	Shampoo, Socks, Car, Soap, Chips, Perfume, Watch, Cream	400000	Male	AUSTRALIA
10	Tom Southwell	488	Cream, Facewash, Mobile, Soap, Watch, Gel, Shampoo	600000	Male	NEW ZEALAND

Figure 8.13: Vendors filtered popular facelist

Chapter 9

Conclusion and Future Scope

9.1 Conclusion

The project will be build keeping into consideration that , it will be helpful for the advertising agencies to get the report of the players according to the sports that the company wants the certain player to promote a product, which can result into huge profit for the company as well as the player.

9.2 Future Scope

The emergence of artificial intelligence means we're better able to understand customer needs. In fact of spending advertising money to remind consumers that brands exist, the brands can view customer needs in real time and focus their messaging on the things that matter to those customers. We can suggest looking to automation and other emerging technologies to deliver products and operational efficiencies that can create the cost savings businesses are looking for, rather than stripping out product features. With the advancement in digital media, it is easy to interact with the sportsperson and also it can help in the advertising industry. By the use of classifier, we can automatically classify the model well suited for the industry. For example, we have an advertisement based on smartphones we can choose a female model who is fair and beautiful. And along with that, we can choose a male model on the basis of budget.

References

- [1] Abhinav Yashkar *Quantifying the performance of players in a football match, IJETT - Volume 45 Number 1- March 2017*
- [2] Ramya Nagarajan, Yihang Zhao, Lin Li *Effective NBA Player Signing Strategies Based on Salary Cap and Statistics Analysis, IEEE 3rd International Conference on Big Data Analysis*
- [3] Avinash Chandra Pandey, Dharamveer Singh Rajpoot, Mukesh Saraswat *Twitter sentiment analysis using hybrid cuckoo search method “,” Information Processing and Management 53 2017*
- [4] Drs Raffaele Poli , Loic Ravenel , Roger Besson *Technical analysis of player performance ” , “ CIES Football Observatory Monthly Report Issue 15 May 2016*

Achievements

I Publications

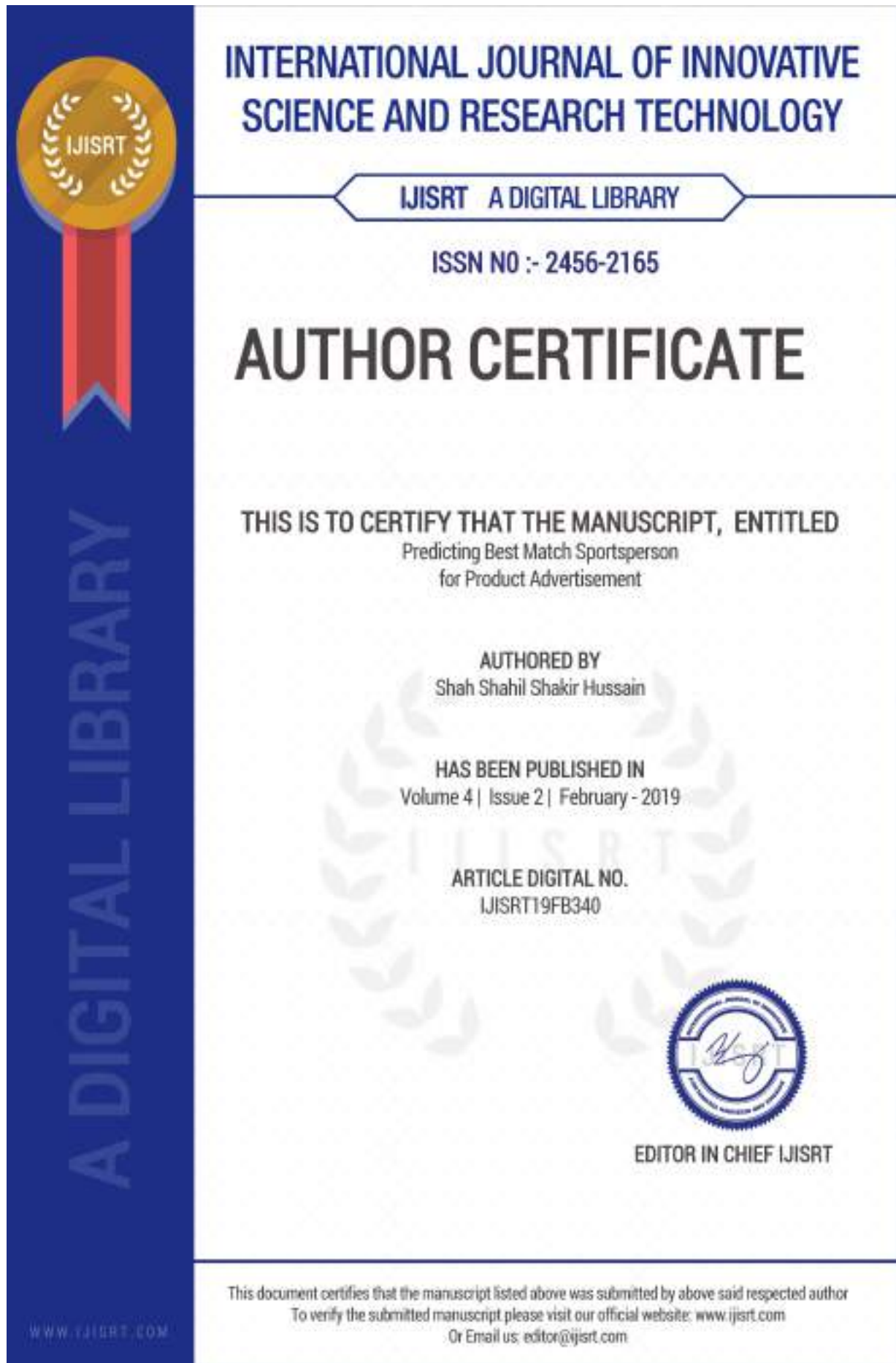
- (a) *Predicting Best Match Sportsperson for Product Advertisement*; Shah Shahil Shakir Hussain Hafizunnisa, Sarguroh Junaid Jawed Sadaf, Gavandi Abhay Audumbar Varsha, Syed Areeb Iqbal Ahmad Rehana, IJISRT, February, 2019 of published(<http://www.ijisrt.com>)


II Conferences

- (a) *Predicting Best Match Sportsperson for Product Advertisement*; Shah Shahil Shakir Hussain Hafizunnisa, Sarguroh Junaid Jawed Sadaf, Gavandi Abhay Audumbar Varsha, Syed Areeb Iqbal Ahmad Rehana, Avalon 2019, March and 2019 of attend(Venue : Venue : Terna Engineering Collge, Nerul)

III Project Competitions

- (a) *Predicting Best Match Sportsperson for Product Advertisement*; Shah Shahil Shakir Hussain Hafizunnisa, Sarguroh Junaid Jawed Sadaf, Gavandi Abhay Audumbar Varsha, Syed Areeb Iqbal Ahmad Rehana, Paper Presentation , March and 2019 of attend(Venue : Terna Engineering Collge, Nerul)





A DIGITAL LIBRARY

WWW.IJISRT.COM

INTERNATIONAL JOURNAL OF INNOVATIVE SCIENCE AND RESEARCH TECHNOLOGY

IJISRT A DIGITAL LIBRARY

ISSN NO :- 2456-2165


AUTHOR CERTIFICATE

THIS IS TO CERTIFY THAT THE MANUSCRIPT, ENTITLED
Predicting Best Match Sportsperson
for Product Advertisement

AUTHORED BY
Sarguroh Junaid Jawed


HAS BEEN PUBLISHED IN
Volume 4 | Issue 2 | February - 2019

ARTICLE DIGITAL NO.
IJISRT19FB340



EDITOR IN CHIEF IJISRT

This document certifies that the manuscript listed above was submitted by above said respected author
To verify the submitted manuscript please visit our official website: www.ijisrt.com
Or Email us: editor@ijisrt.com



A DIGITAL LIBRARY

WWW.IJISRT.COM

INTERNATIONAL JOURNAL OF INNOVATIVE SCIENCE AND RESEARCH TECHNOLOGY

IJISRT A DIGITAL LIBRARY

ISSN NO :- 2456-2165


AUTHOR CERTIFICATE

THIS IS TO CERTIFY THAT THE MANUSCRIPT, ENTITLED
Predicting Best Match Sportsperson
for Product Advertisement

AUTHORED BY
Gavandi Abhay

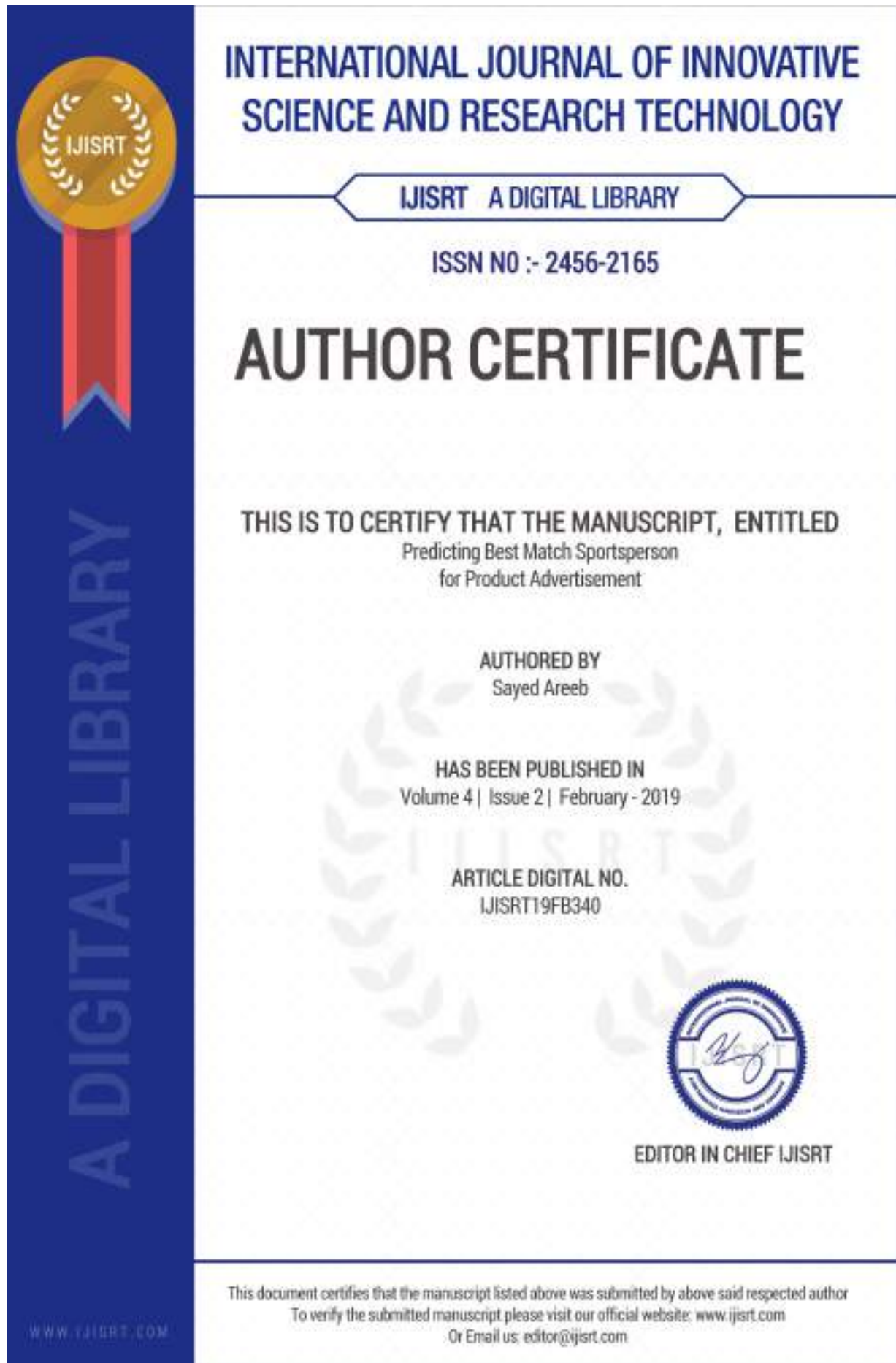
HAS BEEN PUBLISHED IN
Volume 4 | Issue 2 | February - 2019

ARTICLE DIGITAL NO.
IJISRT19FB340



EDITOR IN CHIEF IJISRT

This document certifies that the manuscript listed above was submitted by above said respected author
To verify the submitted manuscript please visit our official website: www.ijisrt.com
Or Email us: editor@ijisrt.com



TERNA PUBLIC CHARITABLE TRUST'S
TERNA ENGINEERING COLLEGE
AN ISO 9001 : 2000 | NBA ACCREDITATION



CERTIFICATE OF PARTICIPATION

This is to certify that

Shah shahil

of

A.T. Kalsekar Technical Campus

has participated in

Avalon 2019, A National Level
(Technical Paper Presentation / Project Competition)
conducted on 5th & 6th March, 2019
at Terna Engineering College, Nerul




Prof. D.M. Bavkar
Avalon co-ordinator


Dr. L.K. Ragha
Principal

TERNA PUBLIC CHARITABLE TRUST'S
TERNA ENGINEERING COLLEGE
AN ISO 9001 : 2000 | NBA ACCREDITATION



CERTIFICATE OF PARTICIPATION

This is to certify that

Junaïd Sargurh

of

A.I. Kalsekar Technical Campus

has participated in

Avalon 2019, A National Level
(Technical Paper Presentation / Project Competition)
conducted on 5th & 6th March, 2019
at Terna Engineering College, Nerul




Prof. D.M. Bavkar
Avalon co-ordinator


Dr. L.K. Ragha
Principal

TERNA PUBLIC CHARITABLE TRUST'S
TERNA ENGINEERING COLLEGE
AN ISO 9001 : 2000 | NBA ACCREDITATION



CERTIFICATE OF PARTICIPATION

This is to certify that

Abhay Gavandi

of

A.I. Kalsekar Technical Campus

has participated in

Avalon 2019, A National Level
(Technical Paper Presentation / Project Competition)
conducted on 5th & 6th March, 2019
at Terna Engineering College, Nerul




Prof. D.M. Bavkar
Avalon co-ordinator


Dr. L.K. Ragha
Principal

TERNA PUBLIC CHARITABLE TRUST'S
TERNA ENGINEERING COLLEGE
AN ISO 9001 : 2000 | NBA ACCREDITATION



CERTIFICATE OF PARTICIPATION

This is to certify that

Syed Areeb

of

A. I. Kalsekar Technical Campus

has participated in

Avalon 2019, A National Level
(Technical Paper Presentation / Project Competition)
conducted on 5th & 6th March, 2019
at Terna Engineering College, Nerul




Prof. D.M. Bavkar
Avalon co-ordinator


Dr. L.K. Ragha
Principal