Alina Baber

[Email address]

Group Member

[Draw your reader in with an engaging abstract. It is typically a short summary of the document.   
When you’re ready to add your content, just click here and start typing.]

PSL Analysis Application

Project Report

Table of Contents

[1 Abstract 4](#_Toc71203106)

[2 Introduction 5](#_Toc71203107)

[3 About PSL App Team 6](#_Toc71203108)

[3.1 Team Members 7](#_Toc71203109)

[4 Project Goals 7](#_Toc71203110)

[4.1 Desktop Application for PSL Analysis Mobile Application of Assignments 7](#_Toc71203111)

[4.2 Desktop Application for PSL Cricket Analysis App with Assignments Deadline Reminder 7](#_Toc71203112)

[5 Rationale: Purpose of pursuing this project and the problem that it could solve. 7](#_Toc71203113)

[5.1 Problem 7](#_Toc71203114)

[5.2 Solution 8](#_Toc71203115)

[5.2.1 User Guide to Cricket 8](#_Toc71203116)

[5.2.2 Team Performance and Player Performance Analysis and Prediction 8](#_Toc71203117)

[6 Finished Product Users: who will benefit from using this application 8](#_Toc71203118)

[7 Required Technologies 8](#_Toc71203119)

[7.1 Visual Studio (Development Software) 8](#_Toc71203120)

[7.2 Python Language (Language) 9](#_Toc71203121)

[7.3 Django Machine Learning 9](#_Toc71203122)

[7.4 Machine Learning Classification Model (Random Forest, SVM, KNN and Voting Classifier) 9](#_Toc71203123)

[7.5 Android Smart Phone (Hardware) 9](#_Toc71203124)

[8 Innovation: Team and Player Performance Prediction and Analysis 10](#_Toc71203125)

[8.1 Do you want guide to Cricket or want to know how to play? 10](#_Toc71203126)

[8.2 Do you want to know this Team is top favorite and how it is performing? 10](#_Toc71203127)

[Step.1 Research and Analysis: 10](#_Toc71203128)

[Step.2 Design and Architecture 10](#_Toc71203129)

[Step.3 Graphic User Interface Development 10](#_Toc71203130)

[Step.4 Library and Dataset Integration 10](#_Toc71203131)

[Step.5 Backend Coding and Development 11](#_Toc71203132)

[Step.6 Api Installation on Herku Server 11](#_Toc71203133)

[Step.7 Testing and Deployment. 11](#_Toc71203134)

[8.3 Why would we use PSL instead of others? 11](#_Toc71203135)

[9 Design and Plan for Application Development 11](#_Toc71203136)

[9.1 Application Architecture Design of PSL Application 12](#_Toc71203137)

[9.2 User Case Diagram of PSL Application 13](#_Toc71203138)

[9.3 UML Sequence Diagram of the PSL Application 14](#_Toc71203139)

[9.4 Web data scrapping 14](#_Toc71203140)

[9.5 Data Processing Machine Learning 15](#_Toc71203141)

[9.5.1 Data Cleaning and Statistical Analysis 15](#_Toc71203142)

[**9.5.2** Treatment of Missing Values 16](#_Toc71203143)

[**9.5.3** Data Transformation and Feature Selection 16](#_Toc71203144)

[9.6 Modelling & Prediction 17](#_Toc71203145)

[9.6.1 K nearest Neighbors 17](#_Toc71203146)

[How to choose the optimal value of K? 19](#_Toc71203147)

[Curse of Dimensionality 20](#_Toc71203148)

[9.7 Modelling & Prediction 21](#_Toc71203149)

[9.7.1 Random Forest 21](#_Toc71203150)

[Decision Tree Learning 21](#_Toc71203151)

[Extra Trees 21](#_Toc71203152)

[Random forest classifier 22](#_Toc71203153)

[Prediction Pseudocode: 23](#_Toc71203154)

[9.8 Model Results 23](#_Toc71203155)

[10 Systems Analysis 28](#_Toc71203156)

[10.1 Final Product 28](#_Toc71203157)

[10.2 Description of Product: 35](#_Toc71203158)

[10.3 Comparison between Our PSL Application and other online PSL applications 35](#_Toc71203159)

[11 System Development Methodology 35](#_Toc71203160)

[12 Prototype and Development Phases 36](#_Toc71203161)

[12.1 Phase 1 (GUI Designing) 36](#_Toc71203162)

[12.2 Phase2 (Machine Learning Prediction) 36](#_Toc71203163)

[12.3 Phase 3 (Django api) 36](#_Toc71203164)

[12.4 Phase 4 (Api publication on Heroku) 37](#_Toc71203165)

[13 Limitations 37](#_Toc71203166)

[13.1 Software limitations 37](#_Toc71203167)

[13.2 Hardware limitations 37](#_Toc71203168)

[14 Plan For testing and Finalizing 37](#_Toc71203169)

[14.1 Testing 37](#_Toc71203170)

[14.2 Finalizing 38](#_Toc71203171)

[15 Time Management 38](#_Toc71203172)

[15.1 PSL Application Time management 38](#_Toc71203173)

[16 Troubleshooting and Problem Solving 38](#_Toc71203174)

[17 Conclusion 39](#_Toc71203175)

[18 References 41](#_Toc71203176)

# Abstract

This project presents an innovative way how the Cricket Lovers can stay up to date about the Pakistan Super league Team and Players performance. The main idea is to apply Cricket Team and Players Data analysis and Prediction technique that means the technology which Data processing and analysis with the help of AI. While we were researching about the technology, we explored an advance technology that we could use to predict the Winner most favorite teams and player performance. The purpose of the development was to explore and use a new technology that can facilitate cricket lovers understand the game and analyze the performance of Team and players. We have developed and accomplished a project that will encourage next students’ study and build new things surround the new technologies. The main function of our project “PSL Cricket Analysis App” is to make app which facilitate cricket fans by predicting the top teams by machine learning, give analysis of players, this functionally after a lot of troubleshooting made by the PSL Analysis Application Team, we could complete only because of team work. After all analysis, researches and feedbacks from the faculty supervisor we complete our project by team work in very efficiently.

# Introduction

PSL Analysis Application is software technique which fetch matches and deliveries data and predict the Team and player performance by machine learning and show on app.

Muthuswamy and Lam [1] anticipated Indian bowlers' success against seven international teams that the Indian cricket community plays most frequently against. They used back propagation group and radial base network feature to predict how many runs a bowler is likely to acknowledge and how many wickets a bowler is likely to absorb a particular ODI match.

WikramaSinghe [2] anticipated batsmen's overall success in test series by using a hierarchical linear model. Barr and Kantor [3] have outlined a framework for determining and deciding on batsmen in limited cricket overs. They specified a new measure P(out) i.e. probability to have out and used a two-dimensional graphic illustration with strike rate on one axis and strike rate on another.

Iyer and Sharda [4] used neural networks to predict players' success by separating batsmen and bowlers into three groups separately – winner, slight and failure. They recommend that the player wants to be included in the crew to play World Cup 2007 based on the amount of times a player has earned different ratings.

By comparing the strengths of the two teams, Jhanwar and Paudi [ 5] predict the outcome of a cricket match. For this they assessed the success of each team's individual players. They developed algorithms to version batsmen and bowlers 'results, where they assess a player's ability by analyzing his overall career output after which his recent performances.

Pandey and Niravkumar [6] forecast the success of players in Cricket's Game Using Machine Learning by predicting the runs they will make.

Kalpdrum Passi and Niravkumar Pandey [7] used a regression model to predict the success of Players in One Day International Cricket Matches according to their role in play.

Stylianos Kampakis and William Thomas [8] used ML to estimate English County's outcome of twenty over Cricket matches using past records from players.

Nischal S [9] estimated the results of the Cricket Matches based on previous match winning probabilities. Sethuraman K and Parameswaran Raman [10] concentrated on learning how to evaluate scoring patterns to develop an Algorithm to predict the outcome in Cricket's game.

Information about the game is an essential task for any sport and similarly for the game of cricket as well. The players’ performance varies on various factors. The team management and the captain select eleven players for each match from the entire squad. Review of different attributes and players’ results is considered to pick the best eleven players.

By scoring runs each batsman contributes and each bowler contributes by taking wickets and awarding minimum runs. This project aims to predict team success based on the player's past records. Acquisition of the players 'results individually and their contribution to the team i.e.

Best batting performance among the batsmen available, best bowling performance among the available bowlers and best all-rounder performance will be a great help in selecting the eleven players. We used the Random Forest Algorithm and Decision Tree classifiers to produce the problem's prediction models. It was found that the Random Forest classifier is the most reliable for the problems proposed.

# About PSL App Team

We are the group of enthusiastic IT Students who are always looking forward to innovate the Technology ideas by which can we facilitate and solve people problems. This time around, CCT has given us 3rd year students an opportunity to build something by using all the skills that we have learnt in past 2 years and represent it to the faculty. This motivates us students to develop something which can help people by using advance technology after figuring out the problem people are facing. For us “PSL App Team”, from the very Initial point, we were up to build something different, something useful and most importantly something practical that can solve some sort of problem or something that can improve the current working systems by implementing the latest technologies. After doing some analysis and research on different technologies and problems. We planned something, when seen from our point of view had the potential to be get more advanced with more accuracy. It is the way converting digital and hand written text assignments images in to editable text. So, we “PSL App Team” have decided to develop an application that will be faster and secure more than ever before.

PSL Team comprises of four members, who will work on this Project:

## Team Members

|  |  |
| --- | --- |
| Team Member | Introduction and Expertise |
|  |  |
|  |  |
|  |  |

# Project Goals

The main goal of developing this Desktop application is mentioned below.

## Desktop Application for PSL Analysis Mobile Application of Assignments

* We wanted to solve the issue of students and people who want converted assignments and documents into editable platform
* We want to explore the AI machine learning based prediction and data sciences techniques.
* We wanted to include the functionality of performance prediction to Mobile Application.

## Desktop Application for PSL Cricket Analysis App with Assignments Deadline Reminder

* We wanted that user specially students can automatically schedule.
* Their assignment deadline reminders by the assignment PSL cricket app and get notification.

# Rationale: Purpose of pursuing this project and the problem that it could solve.

## Problem

Predicting the best players for any match in any sport is nothing but predicting the players’ performance. Each individual player performs differently for every match. Player selection from the dataset provided to the machine learning algorithm is very important in the as exactly 11 players are selected at the beginning of the match and are fixed for the entire match unless in case of injury. However, the substituted players in such cases have limitations. Players’ performance can be predicted by analyzing their past records. Cricket players’ abilities and performance is measured in different ways. Batsmen are measured on the basis of batting average, batting strike rate, number of centuries and so on, while bowlers are measured on the basis of bowling average, bowling strike rate, economy rate etc. The players' recent results, however, is a key factor considered during the selection process. Batsman / bowler are also taken into consideration for forecasting his success in the upcoming match at a given venue.

## Solution

To tackle the issue, the proposed solution is a Mobile Application, which resolve all the above-mentioned problems under single platform. Following are the features to solve the issue.

### User Guide to Cricket

* There will be two functionalities in the application one for new cricket fans who are new in cricket, so the application will share the guide and introduction for them.
* The application will be built on Android for Mobile app and Python for API.

### Team Performance and Player Performance Analysis and Prediction

* Software also perform text analysis like it will show which word is using most in the assignment document, how many words are in assignment document etc.
* There will be two functionalities in the application one for new cricket fans who are new in cricket, so the application will share the guide and introduction for them. The application will be built on Android for Mobile app and Python for API.

# Finished Product Users: who will benefit from using this application

* As mention above that this is Flutter based Android application. Mainly it is design for Cricket fans and New to cricket audience.
* The main purpose of this application is to solve the give guide to cricket fans and new audience of cricket and give them prediction about top teams and top players and their performances.

# Required Technologies

## Visual Studio (Development Software)

We used Android Studio for flutter Application development.

* Because, we wanted to develop the cross-platform application and firstly wanted to release for Android operating system with python based Api.
* Android studio provides a complete development environment of easy flutter mobile app development.
* It will also provide the easy flutter configuration and architecture.
* The main reason of using Android Studio is that we can test as we go. Everything we require is available at one platform.

## Python Language (Language)

Following are the reason we used Python language.

* It’s an Interpreted Language
* One of most popular language.
* Easy to Learn and implement.
* Big community support connected to it.
* object-oriented language which builds projects with a clear structure-based programs and allows code to modify, edit and reuse at low development cost.
* Similar to C, JAVA and other popular languages.

## Django Machine Learning

We used Django to make backend and machine learning technology Apis on python.

* Its and open-source web development platform.
* It both used for apis development as well as web development .
* The Django development platform supports a wide range of web and api application development features, example data science, application model, graphics, and data binding.

## Machine Learning Classification Model (Random Forest, SVM, KNN and Voting Classifier)

* Machine learning classification modelling which has high accuracy due to use of with AI machine learning model like Random Forest , SVM , KNN and Voting classifier dataset based classification learning model.

## Android Smart Phone (Hardware)

A Personal Computer or Laptop is required to install and run Visual Studio

Following are the Specification Required to Install Visual Studio 2019:

|  |  |
| --- | --- |
| **Hardware** | **Specification** |
| Android | 1.8 GHz or faster processor. Quad-core or better recommended |
| RAM | 2 GB of RAM; 8 GB of RAM |
| Operating System Version | Android 8 , 9 , 10 or above. |

# Innovation: Team and Player Performance Prediction and Analysis

## Do you want guide to Cricket or want to know how to play?

* This PSL application will facilitate you to learn about cricket and PSL.
* You can easily get the guide how to play cricket.

## Do you want to know this Team is top favorite and how it is performing?

* You can get the list of team performance in PSL
* And Future Prediction of who will win the most in this cricket event.

### Research and Analysis:

We gathered the all details related Cricket match and player performance analysis we studied some research papers and explored other PSL applications to gather details.

### Design and Architecture

We design use case, flow diagrams and graphic user interface to plan the whole application before its development so things should be clear during development.

### Graphic User Interface Development

We developed Flutter based mobile app f with all the input controls, output controls and displays components.

### Library and Dataset Integration

Before coding and development on backend we installed following libraries for python machine learning modelling and data analysis:

1. numpy~=1.19.5
2. pandas~=1.2.4
3. seaborn~=0.11.1
4. scikit-learn~=0.24.2
5. matplotlib~=3.4.1
6. Flask~=1.1.2
7. Werkzeug~=1.0.1
8. joblib~=1.0.1
9. scipy~=1.6.3
10. sklearn~=0.0

PSL Cricket analysis project and add a matches and deliveries data set files to the project which are needed to train our Machine Learning model.

### Backend Coding and Development

After all the above modelling results, we added code to Django api and coded the top winner team, player performance prediction function by using machine learning modelling and.

### Api Installation on Herku Server

After all the development we published api on Herku serverless server.

### Testing and Deployment.

After all the development we tested it by unit testing on multiple Mobile Application.

## Why would we use PSL instead of others?

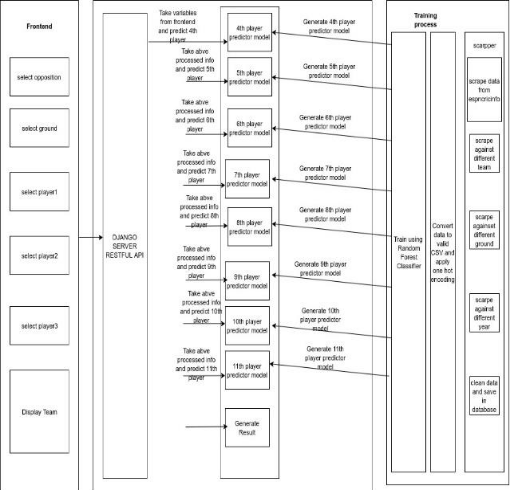
Cricket fans and other people will love to learn about Cricket and want to know the teams and players performances and want to know the score prediction who will win and what will be the performance of team and players.

# Design and Plan for Application Development

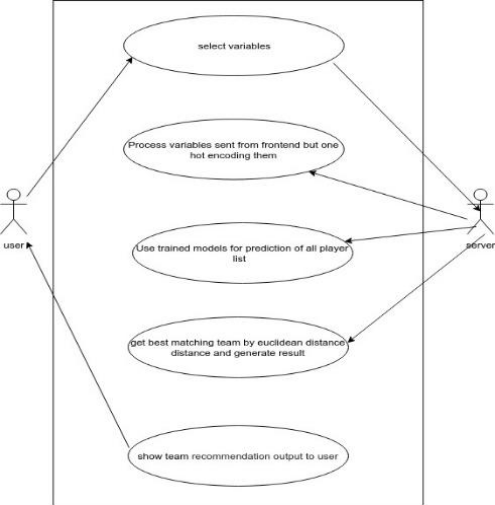
First step includes generating a basic layout of the idea that we have introduced. This layout is going to a simple version of the main application as we don’t want to make things complicated right from the beginning. This basic design will act as a backbone our end goal. In this phase we will focus and talk about two main Areas.

The figure 1 below depicts the architectural block diagram of the proposed system. It basically depicts what features for selection of players will be given to the Django Server and how random forest classifier will be used for prediction of player.

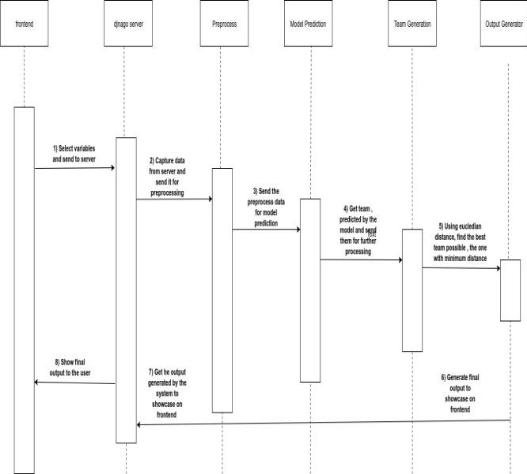
## Application Architecture Design of PSL Application



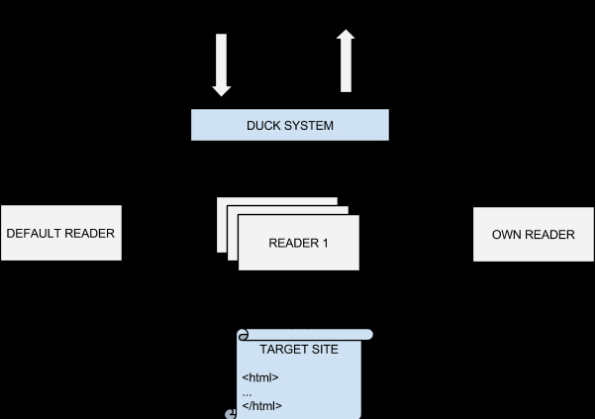
## User Case Diagram of PSL Application



## UML Sequence Diagram of the PSL Application



## Web data scrapping

Web Scraping (Screen Scraping, Web Data Extraction, Web Filtering, etc.) is a method used to retrieve information from websites from which the data is processed to a local file in your laptop or database. Many websites show a lot of useful data but only viewable. There is no provision for the saving for personal use of a copy of that data. The other alternative is to copy and paste the data manually, which is not technically feasible, however, because it can take a lot of time. Web scraping automates this operation, so that data can be easily retrieved without wasting time as shown in figure 4 instead of manually copying the data.  
  


## Data Processing Machine Learning

### Data Cleaning and Statistical Analysis

The process of Data cleaning Is to detect and remove (or correcting) inaccurate records from a dataset, table, or database and refers to recognize unfinished, unreliable, inaccurate or non-relevant parts of the data and then to restore, remodel, or remove the dirty or unwanted data. Data cleaning also performed as batch processing through scripting or interactively with data wrangling tools.

After cleaning, a dataset should be organized as uniform with other related datasets in the operation. The discrepancies identified or eliminated may have been basically applied by user entry mistakes, by corruption in storage or transmission, or by several data dictionary information’s of similar items in several stores. Step are shown in Fig below.

Select desired features and price

Organize dataset and remove outliers

Load Data

### Treatment of Missing Values

Missing data in the training data set can decrease the power / fit of a model or can lead to a biased model because we have not analyzed the behavior and relationship with other features accurately. It can lead to wrong prediction or classification.

There are some reasons for the observations missing in a set of data. Such missing values, although aren’t necessarily wrong but are likely to complicate many techniques of analysis. Missing data and outliers are potentially associated for contributing to high noise and disturbance leading to unreliable prediction in the model. Exploring and finding the missing values and outliers become such an important step that must be conducted before inputting data to a model, otherwise they can gravely impact the data modeling.

Some technique to handle missing values and outliers have their own advantages and disadvantages. There are trade off to consider. Omitting value is easy to perform but the information from an observation is lost; setting values to null is also easy but not every modelling technique implementation can handle null values; imputing a static value such as 0 or the mean do not lose information but can lead to false estimations. Meanwhile, imputing a value from an estimated or theoretical distribution does not disturb the model as much but it can artificially raise dependency among the features.

From the dataset, all features containing missing values or NA’s and outliers were removed to improve the analysis performance. After the data imputations, 1046 observations left. Here we replaced missing values of categorical features with “None” class and missing values of numerical features with median values of the features.

replacing missing values categorical with “None” & Numeric with median value

Checking missing values

Figure shows the missing values map we treated in this process, quantity and ratio in combine 2 columns observation dataset of train and test.

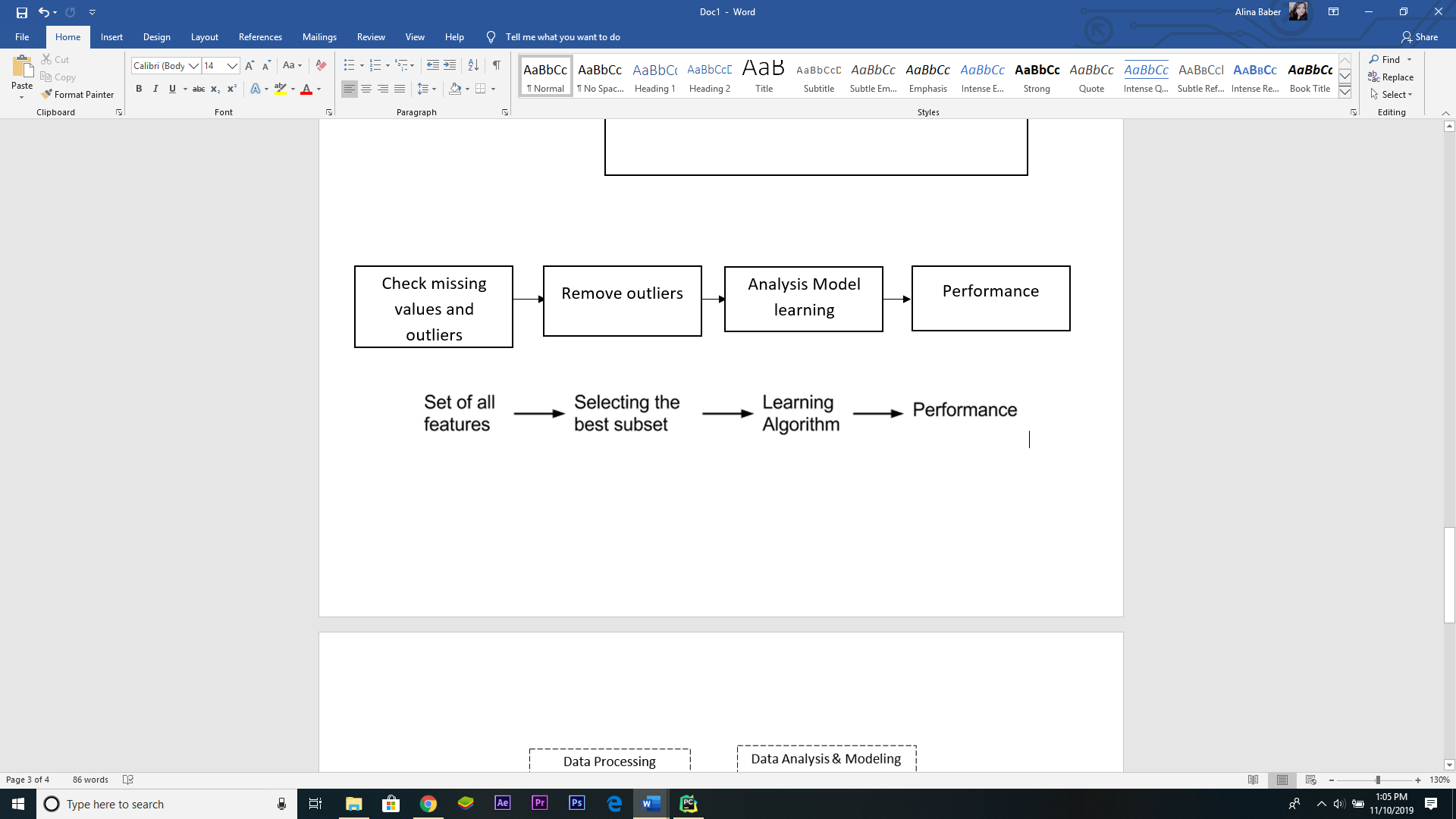
Here we replaced missing values of categorical features with “None” class and missing values of numerical features with median values of the features.

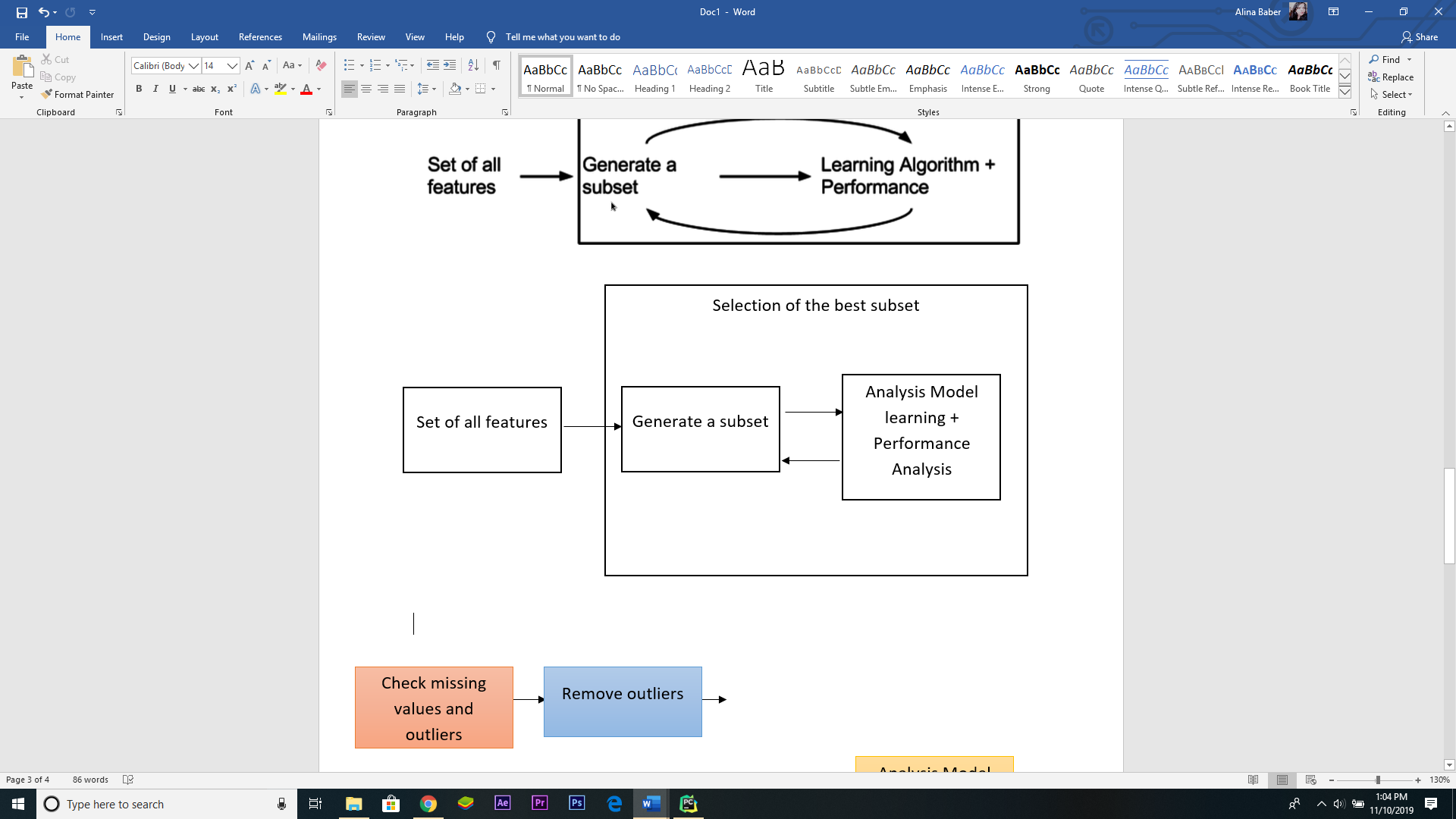
### Data Transformation and Feature Selection

Feature Transformation is a group of techniques that create new features (predictor features). Feature selection is a subset of feature transformation it is done for Knowledge discovery, Interpretability, to gain some insights and Curse of dimensionality, there are two ways of feature selection Filter type techniques select features regardless of the model. They are based only on general features like the correlation with the feature to predict. Filter techniques suppress the least interesting features. They are mainly applied as a pre-process method. Another one Wrapper techniques evaluate subsets of features which allows, unlike filter approaches, to detect the possible interactions between features.

To decrease the number of features, a mixed manual-codes selection technique was employed to select the most important features. explanatory features, features of type numeric or continuous and 5 categorical features were selected so as to simplify the analytical process. The selection of these features was a priority, except the numerical data id, years and months that were considered irrelevant for the analysis. The selected columns remain, and the undesired columns were deleted.

Nevertheless, out of these selected features, further data reduction will only include those features that are significant in the Random Forest model. For this correlation matrix is being Computed for features and highly correlated features are being selected for Initial Random Forest model. Figures below show the all steps to apply.





## Modelling & Prediction

### K nearest Neighbors

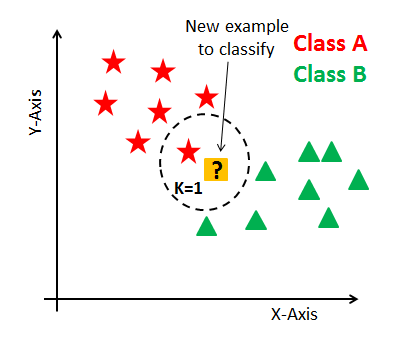
KNN or K Nearest Neighbors is one of the most known classification algorithms as of now in the industry simply because of it is so simple and has high accuracy.

KNN is a simple algorithm that stores all real time cases and classifies new cases based on a similarity measure (e.g., distance functions). KNN has been applied in statistical estimation and pattern recognition from 70s as a non-parametric technique.

The algorithm classifies that similar values occur in close proximity.

In KNN model, K is the number of nearest neighbors. The number of neighbors is the core factor to make decision. **K is generally set as an odd number if the number of classes are 2.** When K=1, in the simple case then the algorithm is defined as the nearest neighbor algorithm.

In figure 14, assume yellow colored “**?**” let's say P is the point, for which label needs to predict. Initially we find the one closest point to P and then the label of the nearest point allotted to P.



then we find the k closest point to P and then classify points by majority votes of its K neighbors. Each object votes for their class and the class with the most votes we get from as the prediction. To search closest similar points, we use to find the distance between points using distance measures such as Euclidean distance, Hamming distance, Manhattan distance, and Minkowski distance. The algorithm has the following basic steps:

1. Compute distance
2. Find closest neighbors
3. Vote for labels



Figure 15 KNN basic steps

Three most commonly applied distance measures applied to compute the distance between point P and its nearest neighbors are represented as :

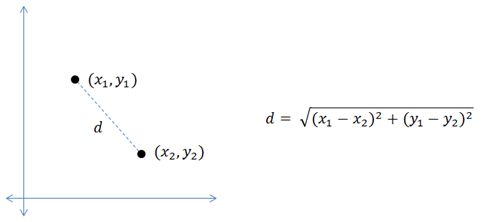
Euclidean distance formula:

Manhattan distance formula:

Minkowski distance formula:

**Euclidean distance:** It is the most commonly applied distance measure also known as simply distance. The application of a Euclidean distance measure is highly recommended by many researches in the case of dense or continuous data. Euclidean distance is the most effective proximity measure. The length of the path between two points is Euclidean distance. The Pythagorean theorem provides this distance between two points.

In Fig below shows how to Compute Euclidean distance between two points in a 2-dimensional plane.



KNN can be applied for both classification and regression predictive problems. However, it is more widely applied in classification problems in the industry. To evaluate any technique, we generally look at 3 important aspects:

1. Ease to interpret the output  
2. Calculation time of the algorithm  
3. Predictive Power

# How to choose the optimal value of K?

The number of neighbors(K) in KNN is a hyperparameter that we need to choose at the time of building wer model. We can think of K as a controlling variable for the prediction model.

Now, choosing the optimal value for K is best done by first inspecting the data. In general, a large K value is more precise as it reduces the overall noise but there is no guarantee. Cross-validation is another way to retrospectively Estimate a good K value by using an independent dataset to validate the K value. Historically, the optimal K for most datasets has been between 3–10. That produces much better results than 1NN(when K=1).

Generally, an odd number is chosen if the number of classes is even. We can also check by generating the model on different values of K and check their performance.

# Curse of Dimensionality

KNN performs better with a lower number of features than a large number of features. We can say that when the number of features increases than it requires more data. Increase in dimension also leads to the problem of overfitting. To avoid overfitting, the needed data will need to grow exponentially as we increase the number of dimensions. This problem of higher dimension is called the Curse of Dimensionality.

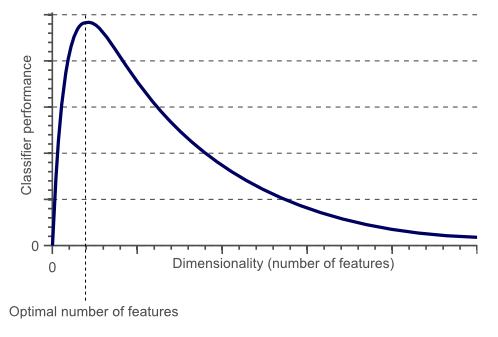


Figure 17 Curse of Dimensionality

From the above graphical representation, it is clearly visible that the performance of wer model decreases with an increase in the number of features(dimensions).

To handle the problem of the curse of dimensionality, we have to perform principal component analysis before applying any machine learning algorithm, or we can also use feature selection technique. According some researches that in large dimension Euclidean distance is not effective anymore. Therefore, we can prefer other measures such as cosine similarity, which get decidedly less affected by high dimension.

**Steps to compute KNN algorithm:**

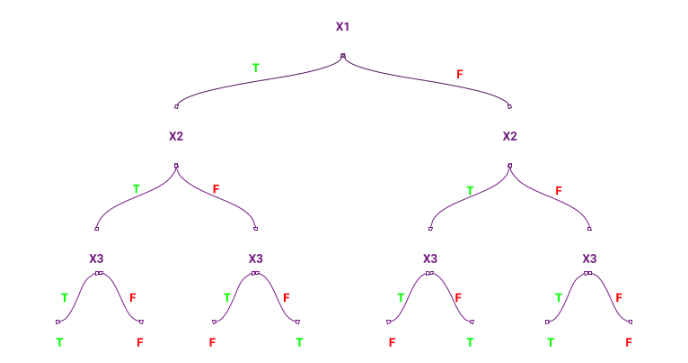
1. Estimate parameter K = number of nearest Neighborss.
2. Compute the distance between the query-instance and all the training samples.
3. Sort the distance and Estimate nearest neighbors based on the Kth minimum distance.
4. Collect the category of the nearest Neighborss
5. Use a simple majority of the category of nearest neighbors as the prediction value of the query.

## Modelling & Prediction

### Random Forest

Random forests or random decision forests are a learning method for classification, regression and other procedures that operate by building a network of decision trees at the time of class (classification) or mean prediction (regression) training and performance mode of the individual trees. Random forests correct the habit of decision trees to over fit to their training range. Preliminaries:

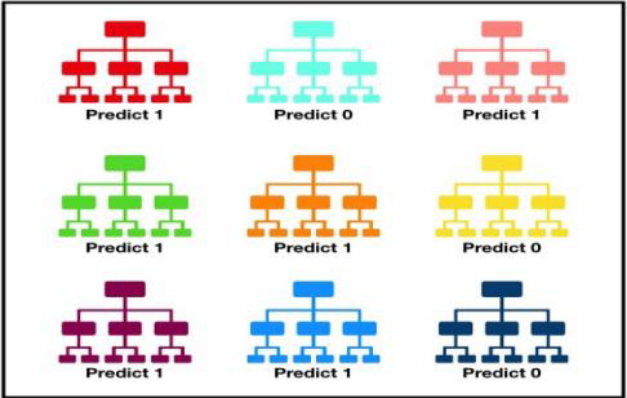
### Decision Tree Learning

The use of Decision trees for various machine learning tasks is a common increase. Tree learning acts as a data mining method, since under scaling it is invariant and many other primary value transformations. This is an element of inspecting models with irrelevant functionality. They're correct though. It is precisely trees which grow very deep that must learn irregular patterns. Random forest summarizes more than one deep decision trees as shown on exceptional grounds in figure 5 with the objective of reducing the variance. However, this does cost a slight increase in bias and some lack of operability, but ultimately improves the output in the final model.  
  


### Extra Trees

Another additional randomization stage generates trees that are highly randomised. Compared to ordinary random forests, there are differences: first, each tree is educated using the entire sample, and second, the top-down splitting is randomized within the tree learner. A random cut-point is chosen instead of determining the near-by cut-factor for every key under consideration. This value is extracted from an even distribution between the appropriate range (in the training set of the tree). Of all the splits generated randomly, the yield that has the highest score is chosen to divide the node. With this parameter, the default values are [root n] with classification, and n for regression, where n is the number of functions in model.

### Random forest classifier

Random forest consists of a huge range of character decision trees that function as an ensemble as depicted in figure 6. Each individual tree within the random forest gives out a category prediction and the elegance with the majority becomes our model’s prediction (see discern below).  
  


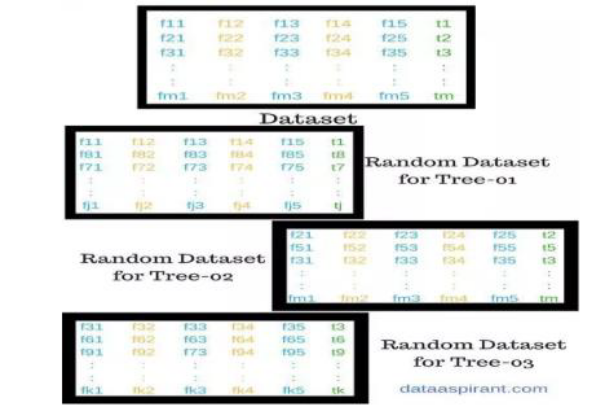
**Working of Random Forest**

The pseudocode for random forest algorithm is as follows: Creation Pseudocode:

Pick "k" from the total features of "m" by chance, where k<=m.

Choose the best split point and measure the "d" node among the functions "k."

Perform node splitting into daughter nodes again using the strongest break.

Repeat the three steps above before hitting "1" number of nodes.Repeat above four steps to develop forest to establish 'n' trees.  
  


### Prediction Pseudocode:

Take the test features and use the rules of will decision tree generated at random to predict the final outcomes and store the predicted outcome (target). Calculate the votes for each of the planned goals. Consider the predicted target strongly voted as the very last prediction from the algorithm of random wooded forest.

## Model Results

The study explores two predictive models of Random forest and k nearest neighbor. Predictive approaches are likely appropriate for situation involving complexity and uncertainty. They will be very useful although it is sometime very hard to model problems in such a way that they could be more convincing modelling than practically useful. Both models make the same set of predictions for the same dataset to come to the outcome values as near as possible. They are ignorant of what nature the predictors are, they matter that the predictors predict well.

However, how well they predict is not as easy as it seems to be although common sense tell the basic test of a predictor is whether it produces predictions that are close to the actual outcomes. It is easy for us to be tempted by a question of comparing the two model of interest in this study if a model predicts more accurate than the other does by merely comparing predictions with outcomes. This research does not intend to justify such a measure of well-predicting nature of the models. Instead, it explores and elaborates the theoretical concepts of the models and the processes taken by their implementations to seek some important interesting points.

The Random Forestmodel is simpler than the k nearest neighbor model in terms of mathematical equations which constitute computational nature the models are trying to solve. When they are implemented in statistical software, which is in this case python, the Random forest results are much easier to interpret than the k nearest neighbors.

The Random forest model results some measures, such as intercept (), coefficient correlations (), residual standard error, adjusted R-squared, F-statistic and probability metric of . On the other hand, such measurements are not assessed in the k nearest neighbors’ architecture. Therefore, what the Random forest has cannot be compared directly and congruently with k nearest neighbors.

Lacking properties for to apple comparison, does not hinder us comparing them in any other ways. For example, the comparison may concern with the predicted values and the actual values of test data. Both Random forest model and k nearest neighbors are capable to provide these predicted values generated by the model based on the same test data. As such, based on these two array values, actual and predicted, we can assess evaluation criterion of Mean Absolute Error (MAE), Root Mean Square Error (RMSE) and Pearson correlation coefficient R. These comparison metrics or criterions are defined by the following mathematical formula:

Where is the actual value and is the predicted value, and is the number of observations.

Where is the actual value and is the predicted value, and is the number of observations?

Where is the actual value and is the predicted value, is the mean of actual value, is the mean of predicted value and is the number of observations.

Figure 22 shows the prediction performance of both models and Table 11 informs the three-comparison criterion defined above. Shows the performance comparison of both algorithms.

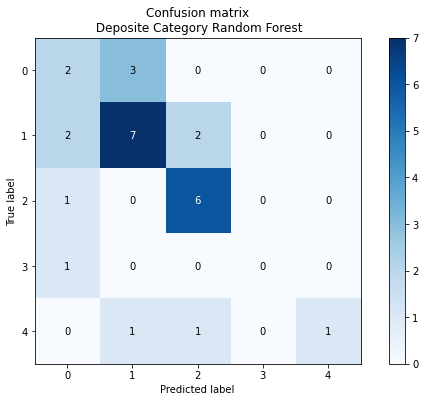




Figure 22 Actual versus predicted of test data by Random Forest& KNN model.

Table 9 Results.

Random Forest Accuracy: 0.4444444444444444

precision recall f1-score support

Islamabad United 0.75 0.43 0.55 7

Karachi Kings 0.50 0.33 0.40 9

Multan Sultan 0.40 0.57 0.47 7

Peshawar Zalmi 0.00 0.00 0.00 1

Quetta Gladiator 0.33 0.67 0.44 3

accuracy 0.44 27

macro avg 0.40 0.40 0.37 27

weighted avg 0.50 0.44 0.45 27

root mean square: 1.6216132799251584

score: -0.7522851919561246

mean absolute error: 1.0740740740740742

mean squared error: 2.6296296296296298

pearson\_coef: 0.2844307321862452

p\_value: 0.15046037574900795

=======================================================================

Accuracy KNN: 0.2962962962962963

precision recall f1-score support

Islamabad United 0.40 0.29 0.33 7

Karachi Kings 0.25 0.11 0.15 9

Multan Sultan 0.30 0.43 0.35 7

Peshawar Zalmi 0.00 0.00 0.00 1

Quetta Gladiator 0.25 0.67 0.36 3

accuracy 0.30 27

macro avg 0.24 0.30 0.24 27

weighted avg 0.29 0.30 0.27 27

root mean square: 1.8856180831641267

score: -1.3692870201096894

mean absolute error: 1.4074074074074074

mean squared error: 3.5555555555555554

pearson\_coef: 0.1300190951741606

p\_value: 0.5180311416516267

=======================================================================

Note: \*) Random Forest & KNN derived from regressing actual vs predicted values.

The Random Forest algorithm outperforms the k nearest neighbors as it is lower in RMSE and MAE & MSE, representing the lower the error the better a model. It is also higher in Score, Pearson correlation coefficient, Mu and Sigma representing the higher the correlation the better a model.

The Random Forest algorithm outperforms the k nearest neighbors as it is lower in all two measurements.

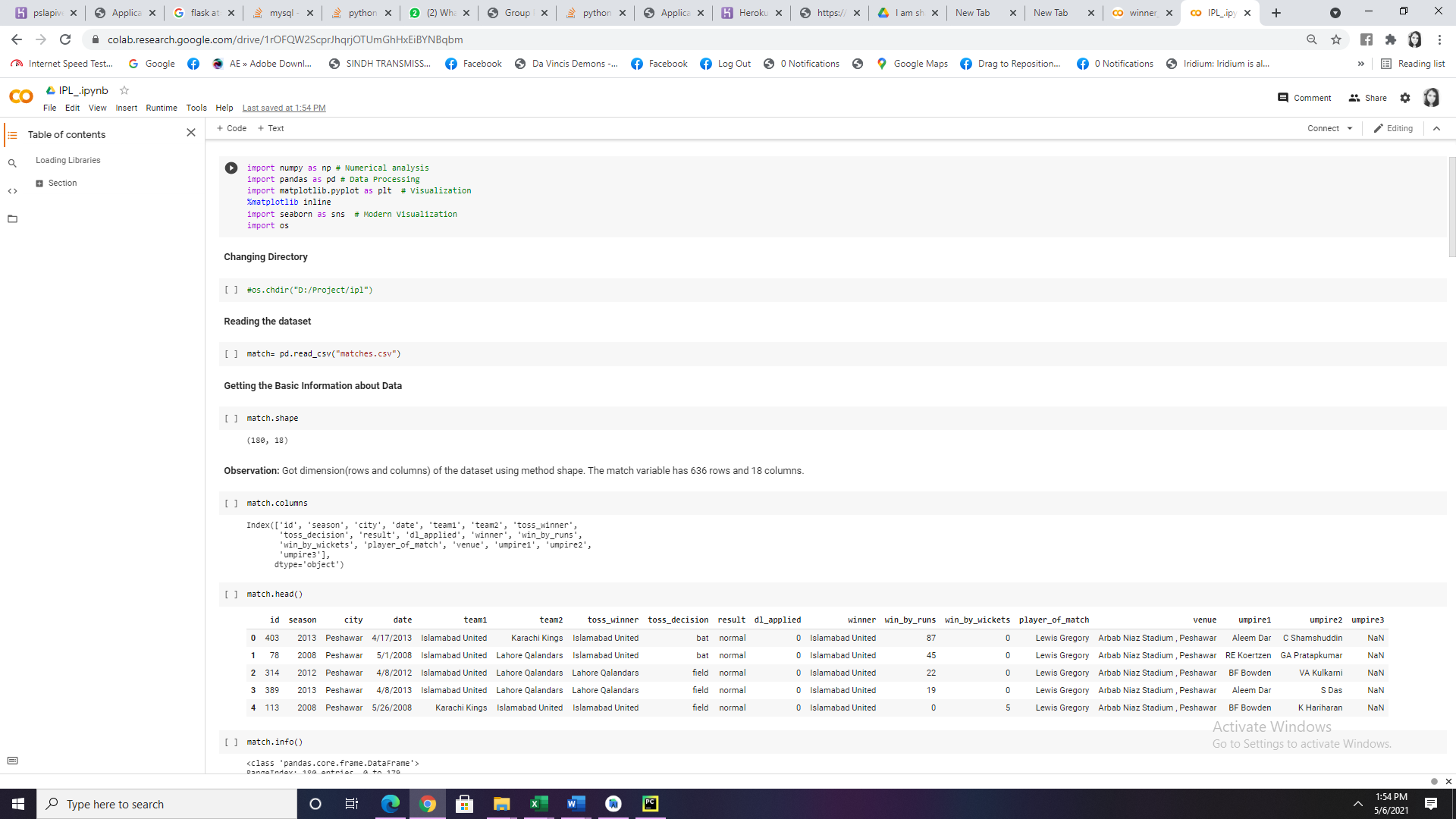
Both algorithms perform fairly well in terms of goodness of fit of the model as indicated by the Score but The Random Forest performed highly correlated to better model, which are 0.90 and 0.58 and in respect to the correlation between actual and predicted variable they reach more than 0.95 Pearson correlation coefficient, which are very strong correlation.

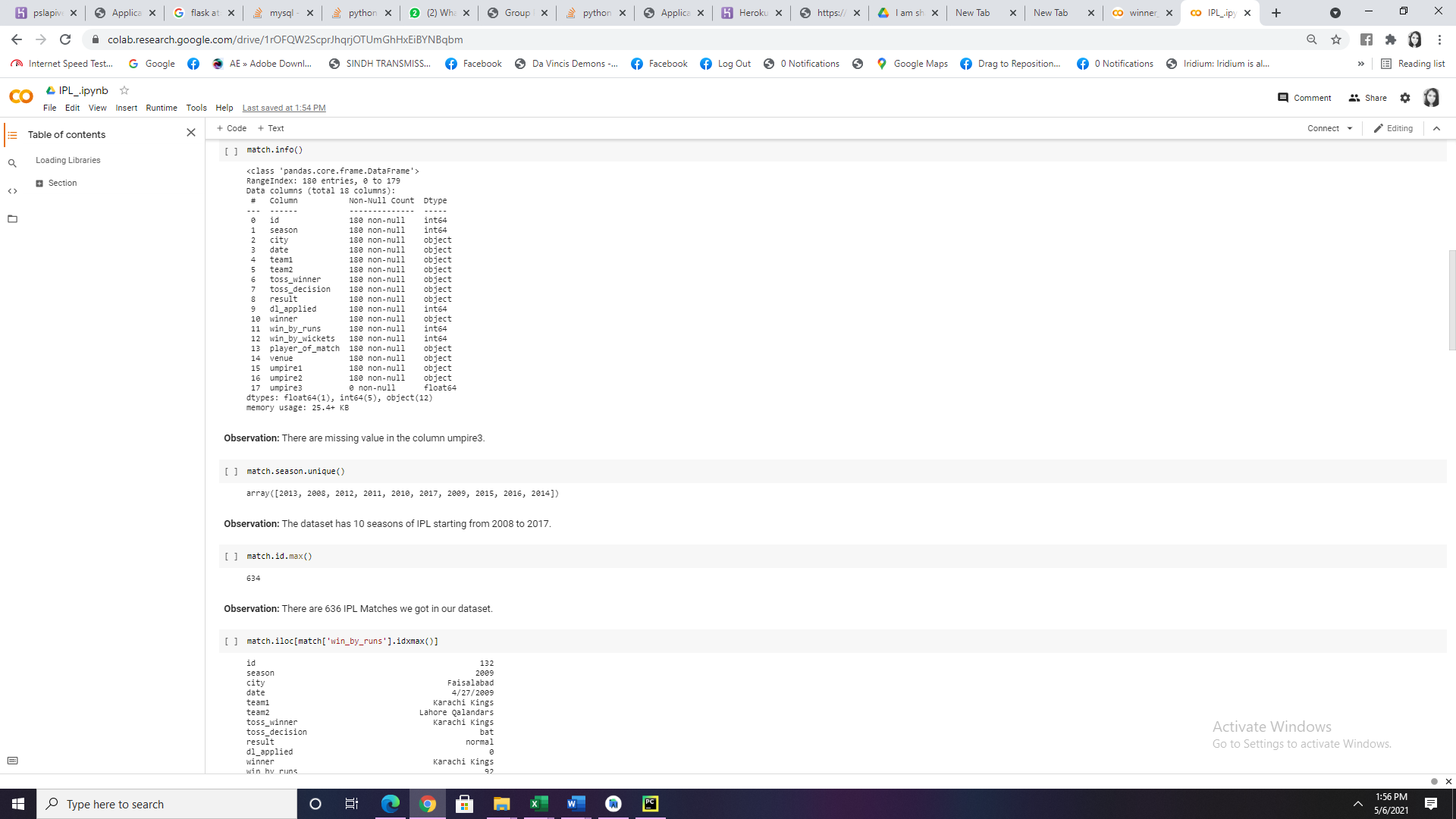
# Systems Analysis

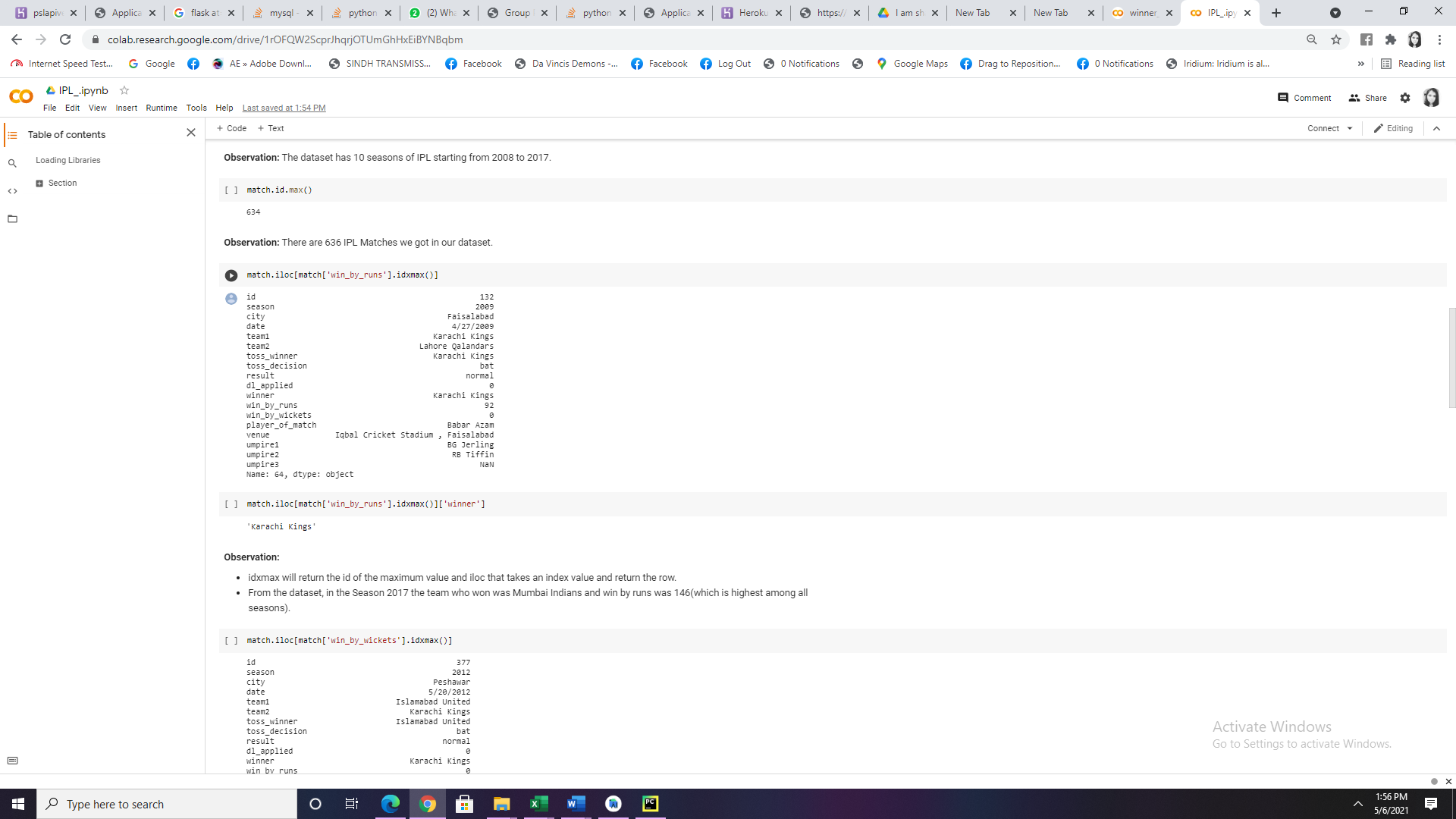
Following is the Final Product of our PSL App with all the features and its comparison with other products

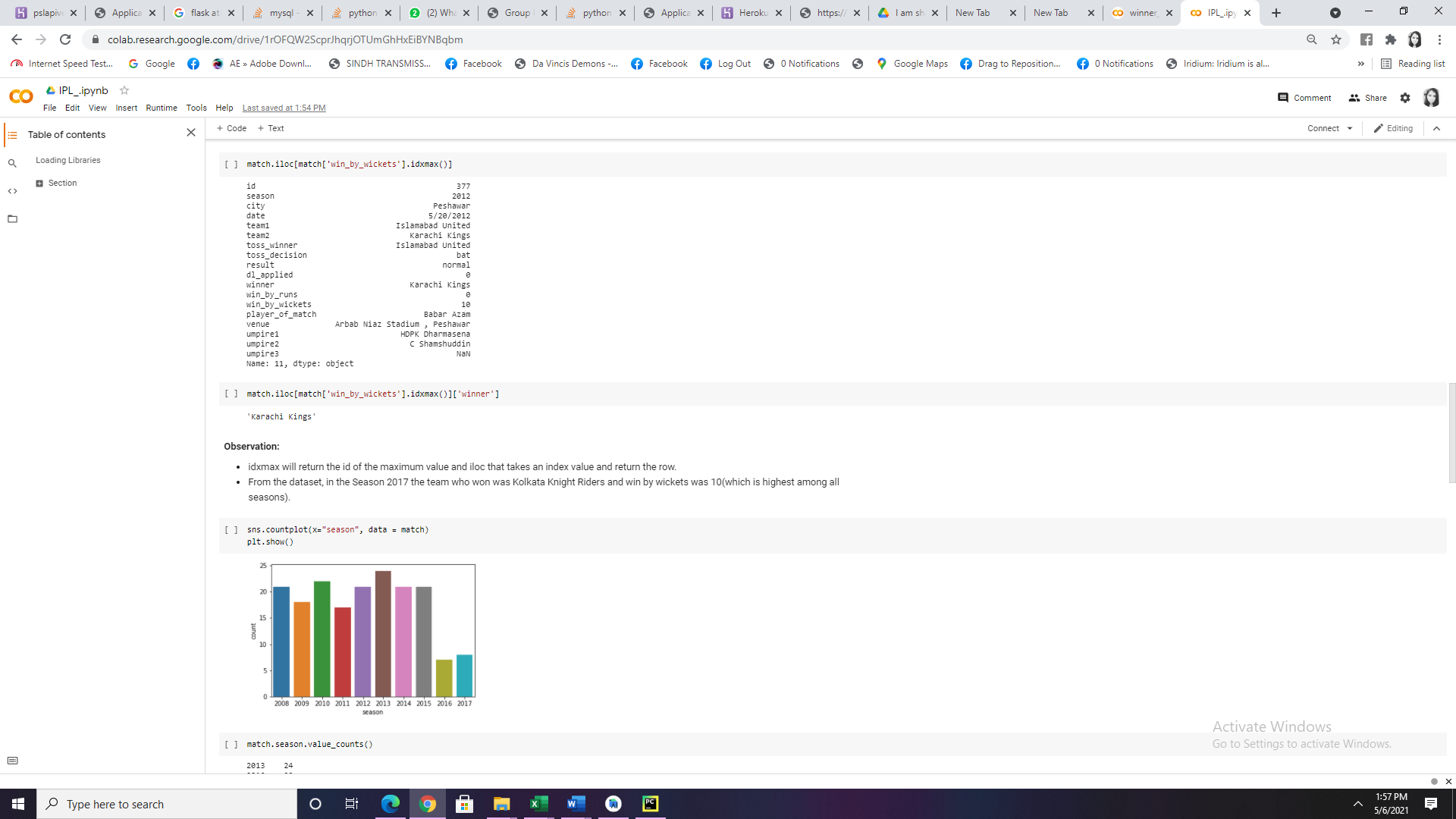
## Final Product

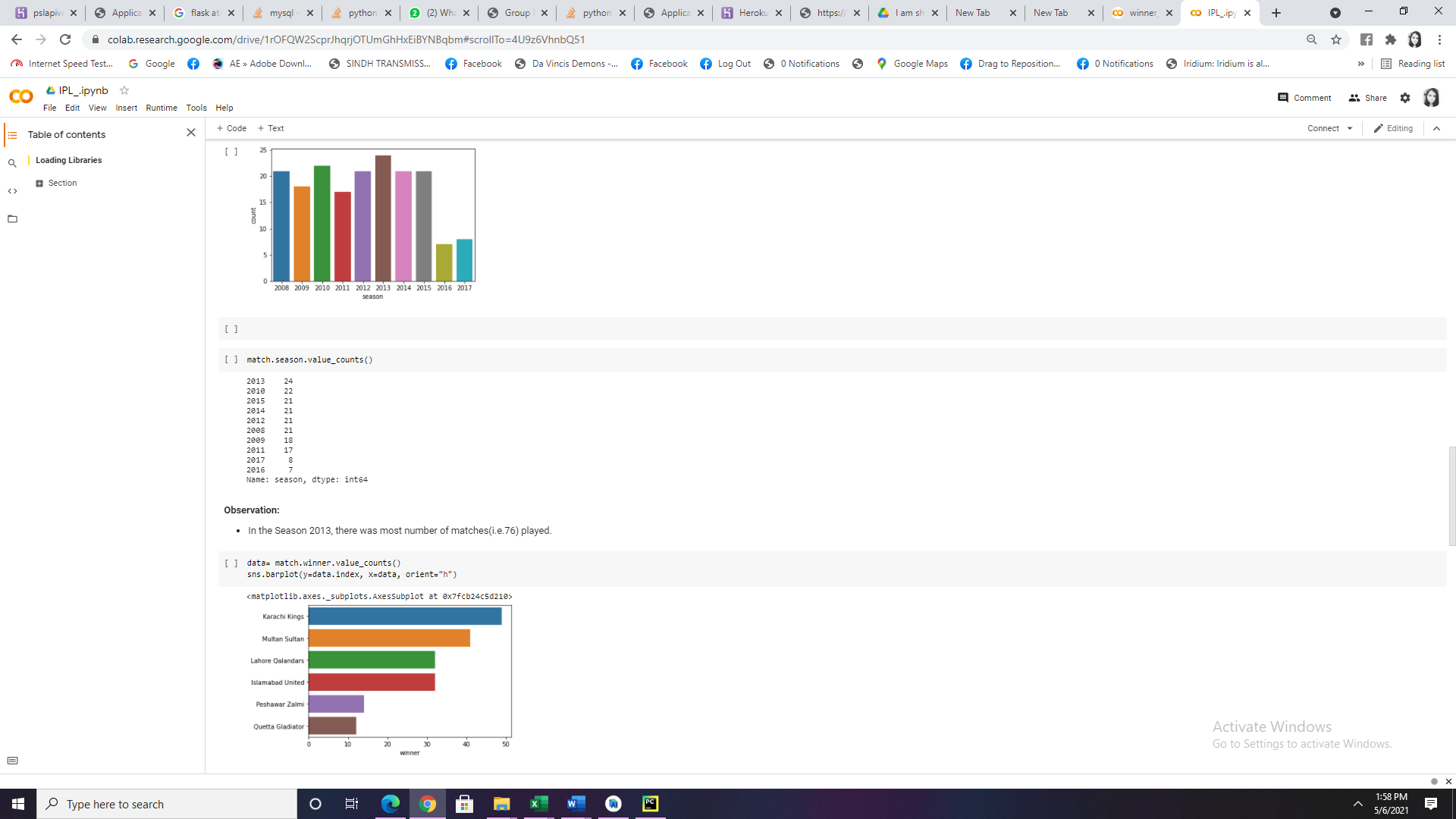
Results:

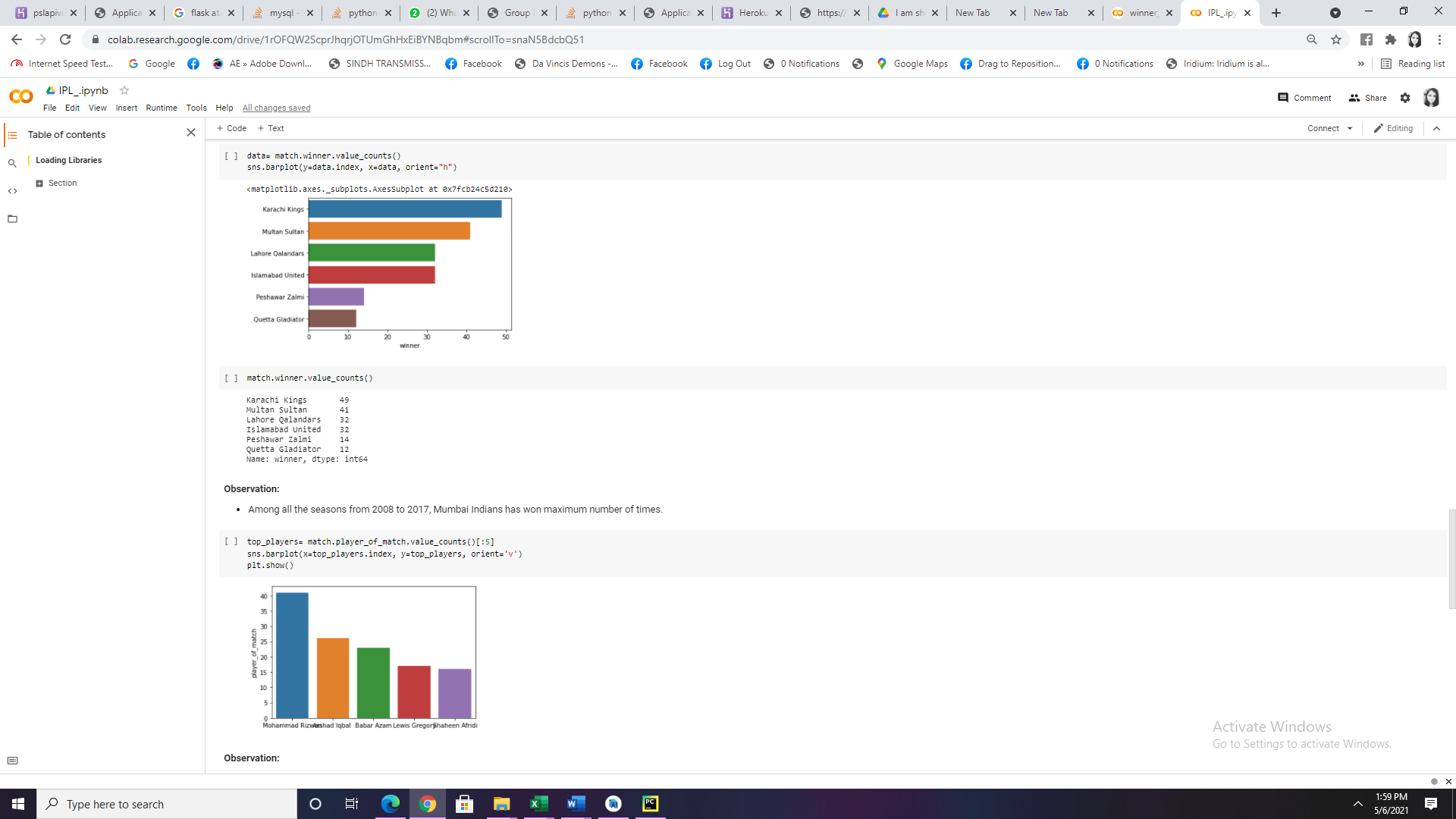


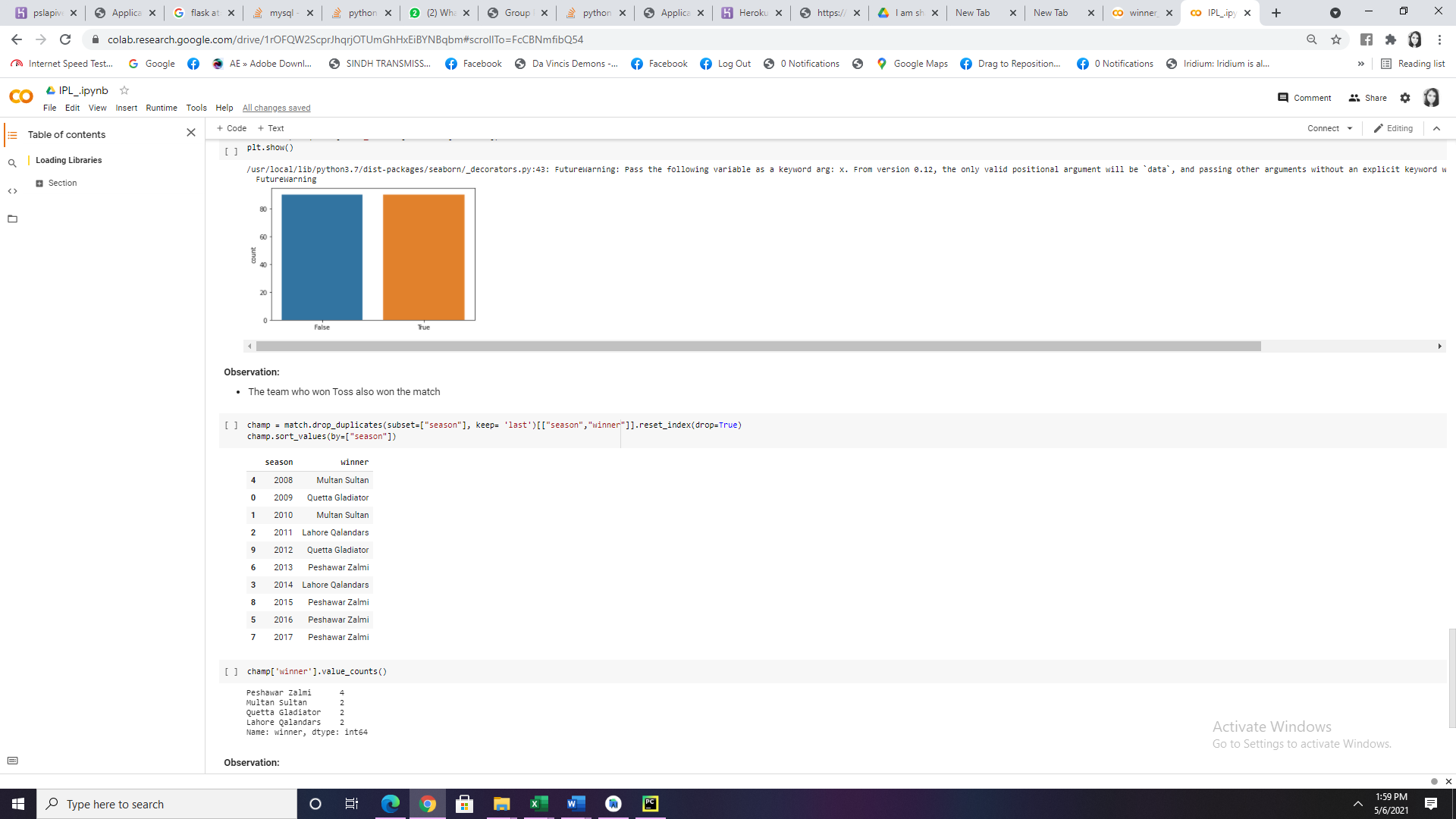


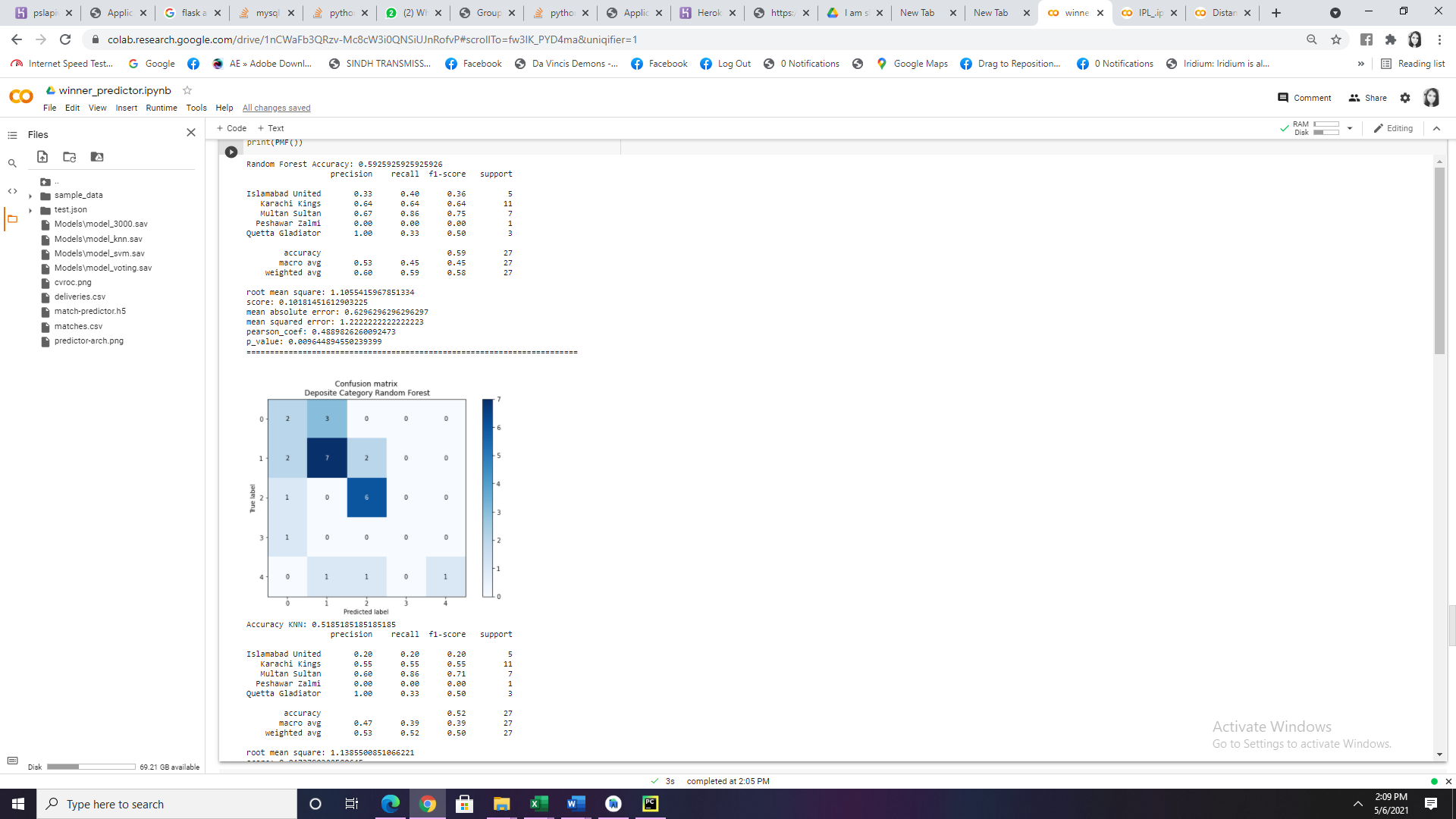


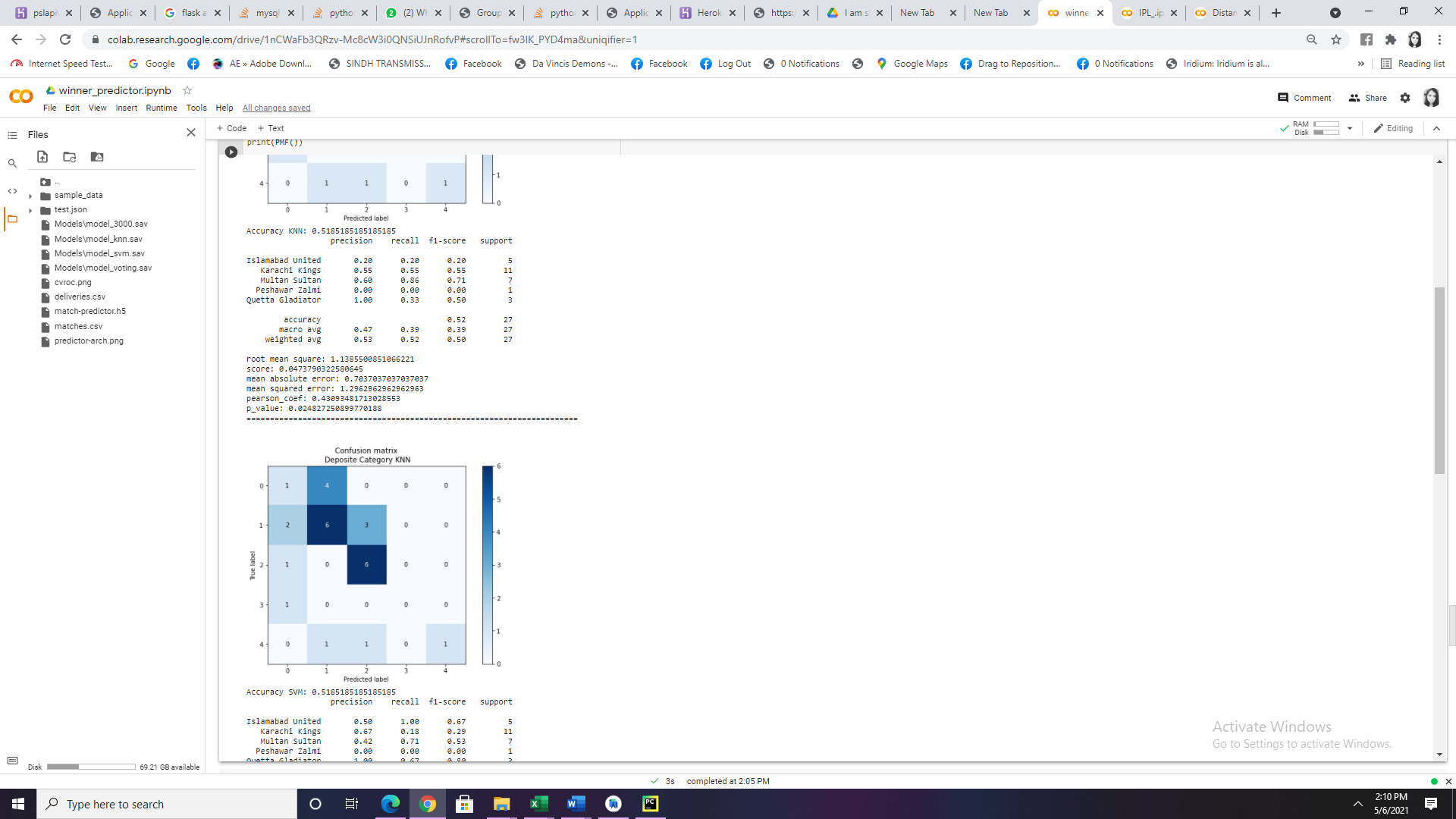












Random Forest Accuracy: 0.4444444444444444

precision recall f1-score support

Islamabad United 0.75 0.43 0.55 7

Karachi Kings 0.50 0.33 0.40 9

Multan Sultan 0.40 0.57 0.47 7

Peshawar Zalmi 0.00 0.00 0.00 1

Quetta Gladiator 0.33 0.67 0.44 3

accuracy 0.44 27

macro avg 0.40 0.40 0.37 27

weighted avg 0.50 0.44 0.45 27

root mean square: 1.6216132799251584

score: -0.7522851919561246

mean absolute error: 1.0740740740740742

mean squared error: 2.6296296296296298

pearson\_coef: 0.2844307321862452

p\_value: 0.15046037574900795

=======================================================================

Accuracy KNN: 0.2962962962962963

precision recall f1-score support

Islamabad United 0.40 0.29 0.33 7

Karachi Kings 0.25 0.11 0.15 9

Multan Sultan 0.30 0.43 0.35 7

Peshawar Zalmi 0.00 0.00 0.00 1

Quetta Gladiator 0.25 0.67 0.36 3

accuracy 0.30 27

macro avg 0.24 0.30 0.24 27

weighted avg 0.29 0.30 0.27 27

root mean square: 1.8856180831641267

score: -1.3692870201096894

mean absolute error: 1.4074074074074074

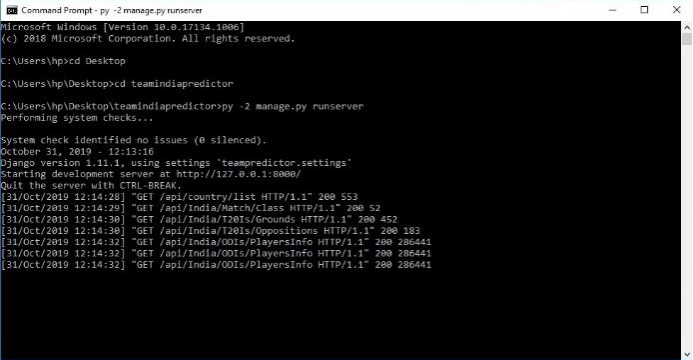
mean squared error: 3.5555555555555554

pearson\_coef: 0.1300190951741606

p\_value: 0.5180311416516267

=======================================================================

Backend:

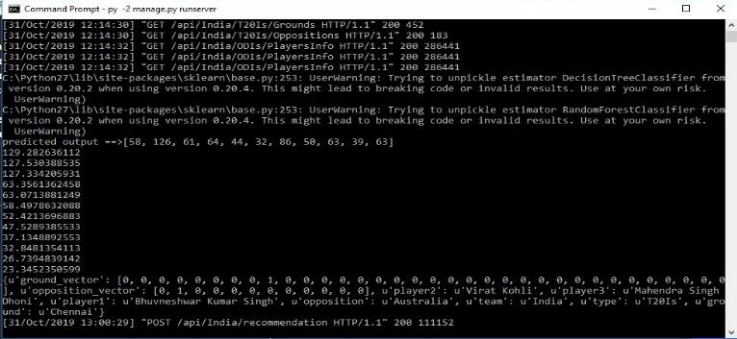


The ‘manage,py’ file is run to initiate the server. All the URL mappings are checked and then the server connects the frontend and backend. It is by default run on port 127.0.0.1:8000The user has to select the Opposition Team from the list as it is an important parameter to analyze the records and predict the team.

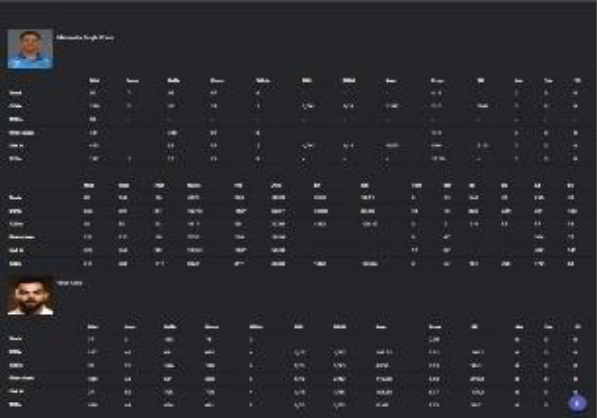
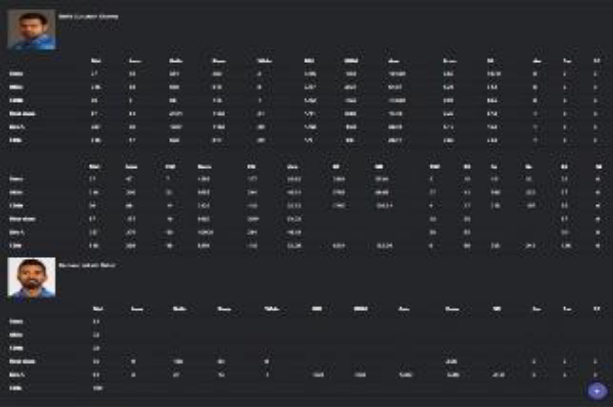
The user has to also select the venue of the match as it classifies if the match is a home match or a away match for the team. It also considers the ground conditions based on the past performances.Every team has some pre-decided star players, which are always sure to be in team, so we have kept the number to three so that we can predict remaining 8 members of the team. Also some base is required for the prediction to happen; these three players will form that base which will give our system a prediction.

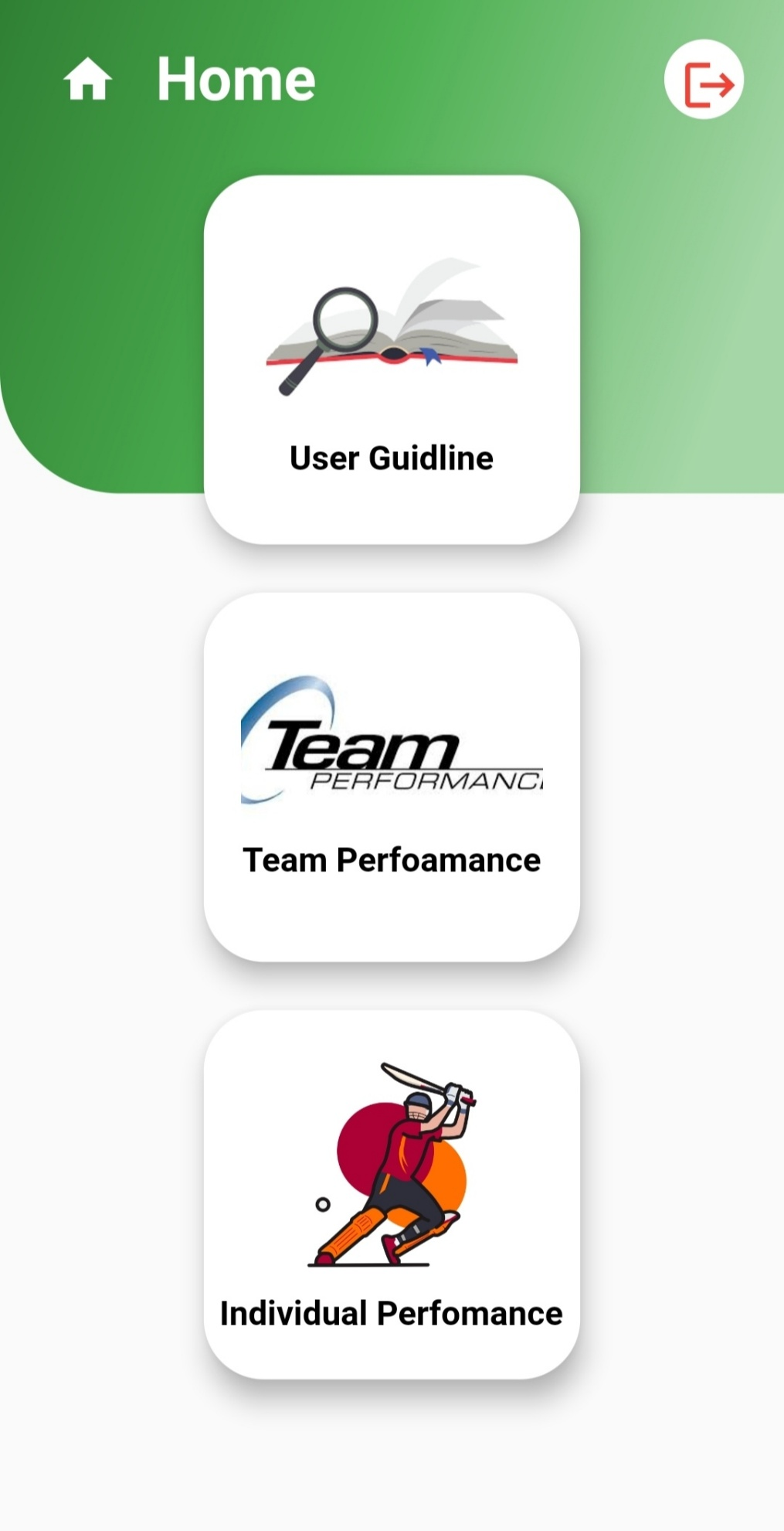
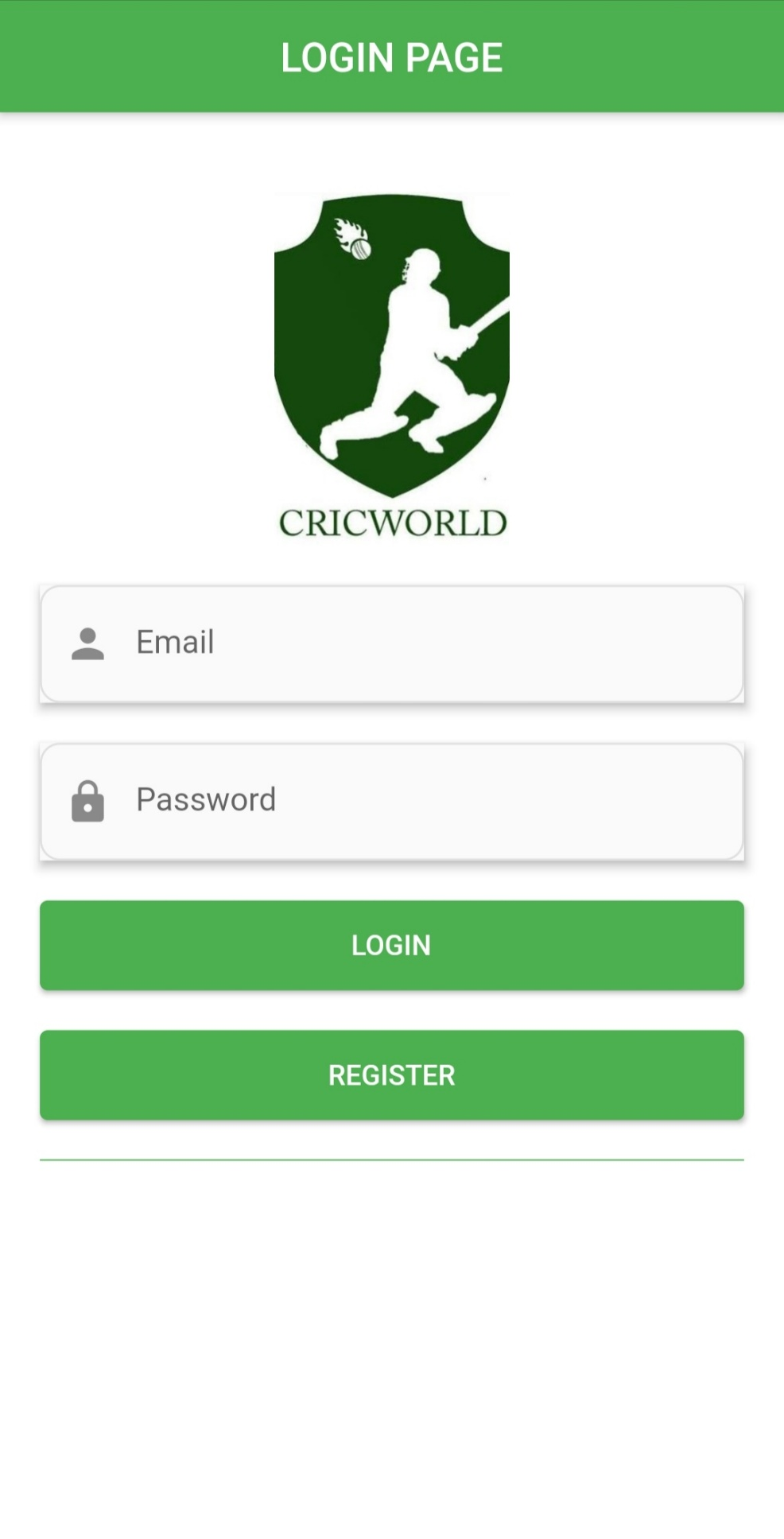
Our smart system is good in predicting team players, based on past records, so more the inputs better is the prediction accuracy.

**Prediction Process**

  
**Predicted Team**

The system gives a team of 11 players based on their past records and considering the opposition team, venue etc.

Below are few examples of the format how the recommended players and their stats are displayed as shown in figure 13.  
  




## Description of Product:

* User has to register and login.
* System will display option to of user guide to cricket, team performance, player performance and top teams & players.
* Application will display the Prediction results of team and player performance and prediction model.

## Comparison between Our PSL Application and other online PSL applications

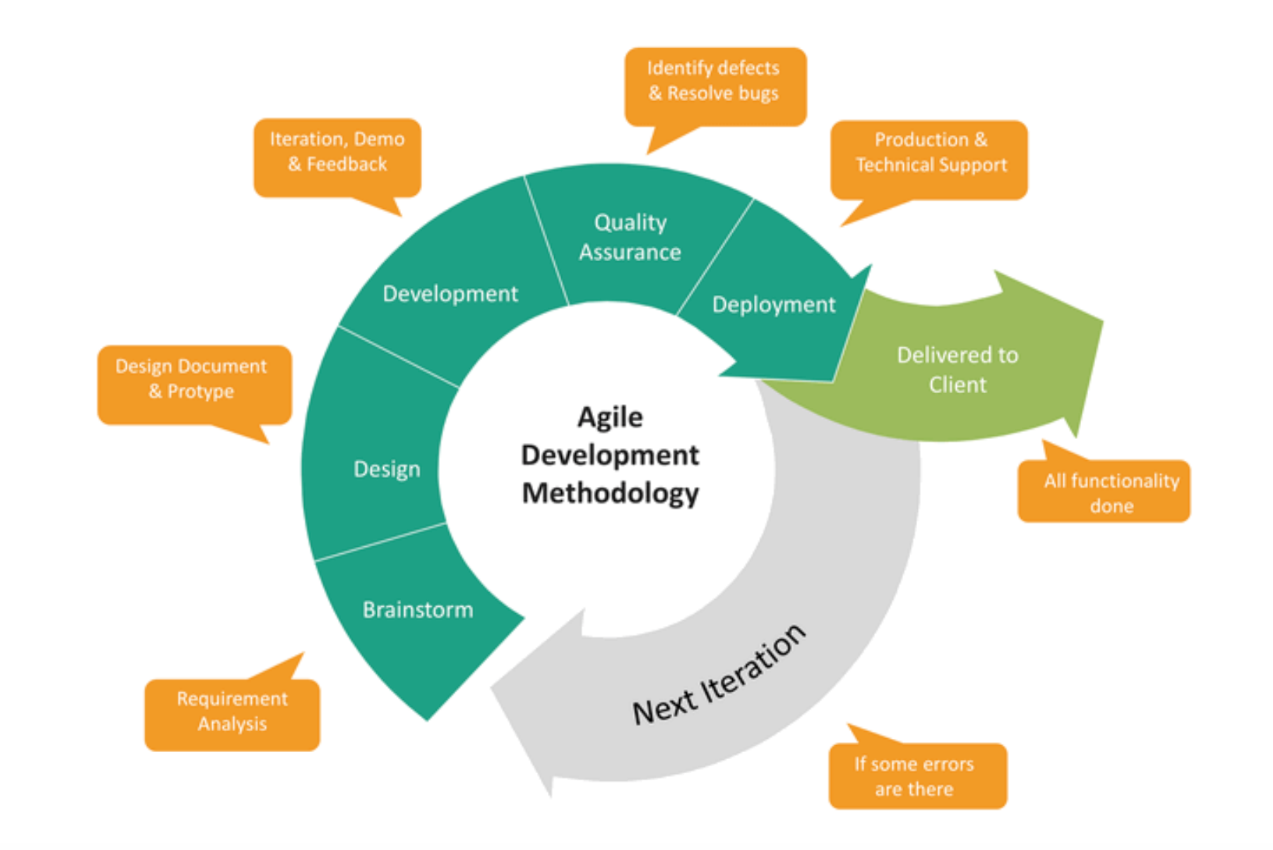
Following are features comparison between Our PSL Application and other online PSL.

|  |  |  |
| --- | --- | --- |
| **Features** | **Our AI based PSL analysis Application** | **Other Online PSL analysis Application** |
| User guide | Available | Available |
| Team Performance | Available | Available |
| Player Performance | Available | Available |
| Top Winner list Prediction | Available | Not Available |
| Cost | Cheaper no subscription or membership needed | Need subscription and membership renewal |

# System Development Methodology

As a team we adapted the Software Life cycle from the Agile approach to build the PSL analysis Application.

* It is a methodology that allowed us to implement the design and also as a guide for our team as an iterative and incremental delivery of our product.
* This strategy is often referred to as “an agile software development,” which its main goal was using an empirical process that allowed us to respond rapidly, efficiently, and effectively to fixed time and cost issues to adapt our needs to the requirements we had to follow while developing our Application.
* Following is the agile development cycle flow diagram.



# Prototype and Development Phases

Following are the agile software development phase we took to complete this project.

## Phase 1 (GUI Designing)

In first phase We developed flutter Mobile application Graphic user interface for our application with all the input controls, output controls and displays components. and get feedback from our supervisor.

## Phase2 (Machine Learning Prediction)

In second phase, we coded machine learning classification of team and player performance by adding library and training dataset. And connected to GUI input controls and out display. And tested it and get feedback from our supervisor.

## Phase 3 (Django api)

In Third phase, we coded backend Api by Django for connecting with mobile application.

## Phase 4 (Api publication on Heroku)

In Fourth phase, we uploaded Django project on Heroku serverless server and published it to connect with mobile application

# Limitations

## Software limitations

This application can be only used on any machine you need to have at least the following software installed:

* Flutter Mobile Application, Android and IOS
* Django, python

## Hardware limitations

* There are no minimum hardware requirements for Android Application rather for the Operating system which should be between Android 8 , 10 or above.
* All the requirements depend on the complexity and effects used in our application which are available in all kind off computers.

# Plan For testing and Finalizing

## Testing

Testing is one of the post necessary phase that is required to see whether that application will work properly in for real time or not. The PSL application go through several tests:

The Mobile application is communally used by a single user who use Android phones or iOS. So, because of it we have specific environment to test and monitor our applications results.

Following are the testing techniques we applied to test our desktop application:

* **Functionality testing:** In this we tested does all the functions we include are working or not.
* **GUI feature testing:** In this we check do all the components and controls are properly integrated and working and displaying the results.
* **Load testing:** In this we check how much image load and data load the application can handle at a time.
* **Backend (directories, datasets and variables) testing:** In this we tested do all the attached files, dataset files, directories are properly attached and responding properly or not.
* **Memory leaks defect testing:** In this testing we tested does our application is memory and time optimized and use optimized way to use less memory space.
* **Compatibility testing:** In this testing we tested which platforms and systems are compatible to run our application.

## Finalizing

This is going to be the last phase, where application is crossed checked and deployed for installation all the small details or errors of the application will be fixed and finalized in this phase.

# Time Management

## PSL Application Time management

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Development Steps | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Research & Analysis | ✔ |  |  |  |  |  |  |  |  |  |
| Design & Architecture |  | ✔ | ✔ |  |  |  |  |  |  |  |
| Development |  |  |  | ✔ | ✔ | ✔ |  |  |  |  |
| Testing |  |  |  |  |  |  | ✔ | ✔ |  |  |
| Finalizing |  |  |  |  |  |  |  |  | ✔ |  |
| Deployment |  |  |  |  |  |  |  |  |  | ✔ |

# Troubleshooting and Problem Solving

As all the software application development requires to follow the steps and in which we face so many issues and challenges. Following are the problems that occurred in the development and testing phase:

|  |  |  |
| --- | --- | --- |
| Phase | Problem | Troubleshoot & Solution |
| Design and Architecture | It was quite challenging to include PSL which can predict both team and player performance so there was dataset required for both | We designed it that one modelling of dataset and predict my machine learning. |
| Dataset Gathering | Dataset of matches and deliveries was another big challenge | We searched the relevant projects and included the dataset to our project which are configurable to our machine learning model requirement. |
| Library Selection and Integration to Project | It was hard to choose and decide which machine learning is more accurate. | We chose Random forest and knn according to our AI based modeling requirement because it use classification modelling technique to train model by dataset. |
| Graphic User Interface designing | We want to design a Mobile application for both Android and iOS. | We developed it on flutter so we can publish it for both Android and iOS. |
| Backend coding and development | These are so many controls and displays in this application as well as so many functionalities so it was challenging to code in under stable way with less memory space and time usage | We used Object oriented Programming architecture of python Django to code it so all the functionalities are understandable, time and memory optimized as well as we used standard variables to make to clear. |
| Testing of Results and Bug fixing | Testing has to be implemented to check all functionalities for both backend and mobile application. | We created standard format for rules for this application to test both backend and mobile application. |
| Deployment | We wanted to Install it to every android phone. | We used Android Studio for flutter deployment kit to both old and latest versions of Windows operating systems. |

# Conclusion

In conclusion, the selection of the right players for any match plays a significant role in the game outcome. A good estimate of the runs a batsman scores, and how many wickets a bowler is likely to take in a match, lets team management pick the best players for each match. Such findings, however, could only be extracted from the data that would come from various sources. We modelled the 11 members team in this project based on the stats and characteristics of the players. Different other factors that influence players' performance could be known as weather, or the design of the ground that could not be included in this study due to data unavailability. Random Forest and KNN proved to be the most accurate classifier with optimum precision for the datasets. Different tests can be performed for all game formats. The models for such formats can be designed to reverse the cricketers' vital characteristics; e.g. batsmen need stamina and the ability to play longer innings in test matches, while in T20 matches score more runs in fewer overs. Bowlers will also have better wicket-taking skills in test matches as well as a higher economy rate, which means they will concede less runs in T20 matches. In addition, attempts can be made to boost classifier accuracies and include other parameters such as location wise results, opponent team, ground conditions, player health, match fees etc. Match fees etc.

# References

[1] S. Muthuswamy and S. S. Lam, "Bowler Performance Prediction for One-day International Cricket Using Neural Networks", Proceedings of IIE Annual Conference and Expo 2008, pp.1391-1395/

[2] I. P. Wickramasinghe, "Predicting the performance of batsmen in test cricket", Journal of Human Sport & Exercise, vol. 9, no. 4, pp. 744-751, May 2014.

[3] G. D. I. Barr and B. S. Kantor, "A Criterion for Comparing and Selecting Batsmen in Limited Overs Cricket", Operational Research Society, vol. 55, no. 12, pp. 1266-1274, December 2004.

[4] S. R. Iyer and R. Sharda, "Prediction of athlete’s performance using neural networks: An application in cricket team selection", Expert Systems with Applications, vol. 36, pp. 5510-5522, April 2009.

[5] M. G. Jhanwar and V. Pudi, "Predicting the Outcome of ODI Cricket Matches: A Team Composition Based Approach," in European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECMLPKDD 2016 2016), 2016.

[6] Pandey, Niravkumar, “Predicting Players’ Performance in the Game of Cricket Using Machine Learning” in Master of Science

[7] Kalpdrum Passi and Niravkumar Pandey, “Predicting Players' Performance in One Day International Cricket Matches Using Machine Learning”, Department of Mathematics and Computer Science, Laurentian University, Sudbury, Canada

[8] Stylianos Kampakis and William Thomas “Using ML to Predict the Outcome of English County twenty over Cricket Matches”, University College London, 2015

[9] Nischal S “Predcting the outcome of Cricket Matches using AI”, November 2017. https://medium.com/@xtaraim/predicting-the-outcome-of-cricket-matches-using-ai-14f61aed7b97

[10] Sethuraman K, Parameswaran Raman, “Learning Algorithm for Prediction in the game of Cricket”, https://people.ucsc.edu/~praman1/static/pub/ML\_Project\_CS7641.pdf