Score Prediction

This project predicts the outcome of an English premier league match in terms of percentage.

TITAN COMPANY LIMITED

Yemlur, HAL Airport Road,
Bengaluru

4th AUGUST 2015

Project On:

Determining Match Results Of Manchester United

Based On Statistics

- Guided By: Mr Venkatesh R

  Senior IT Manager
  TITAN COMPANY LIMITED

- Prepared By: Suryadeep Chatterjee

  Student, B-Tech, 7th Semester
  Sikkim Manipal Institute of Technology

Table of Contents

(II) Requirements..... 26

(III) Design..... 39

(IV) Test Plans

List of Database Tables:

SECTION: - 0(ZERO)

- This section contains a small description about the dynamic database table that has been used to retrieve the data required to perform the calculations in the application.

- The database is created using **MySQL** query language.

- Name of the database is: **rseason.**

- This **rseason** database contains three tables to store different data used in the web application.

- Name of tables in this database is:

  (i) statis
  (ii) lseason
  (iii) auth

NOTE:

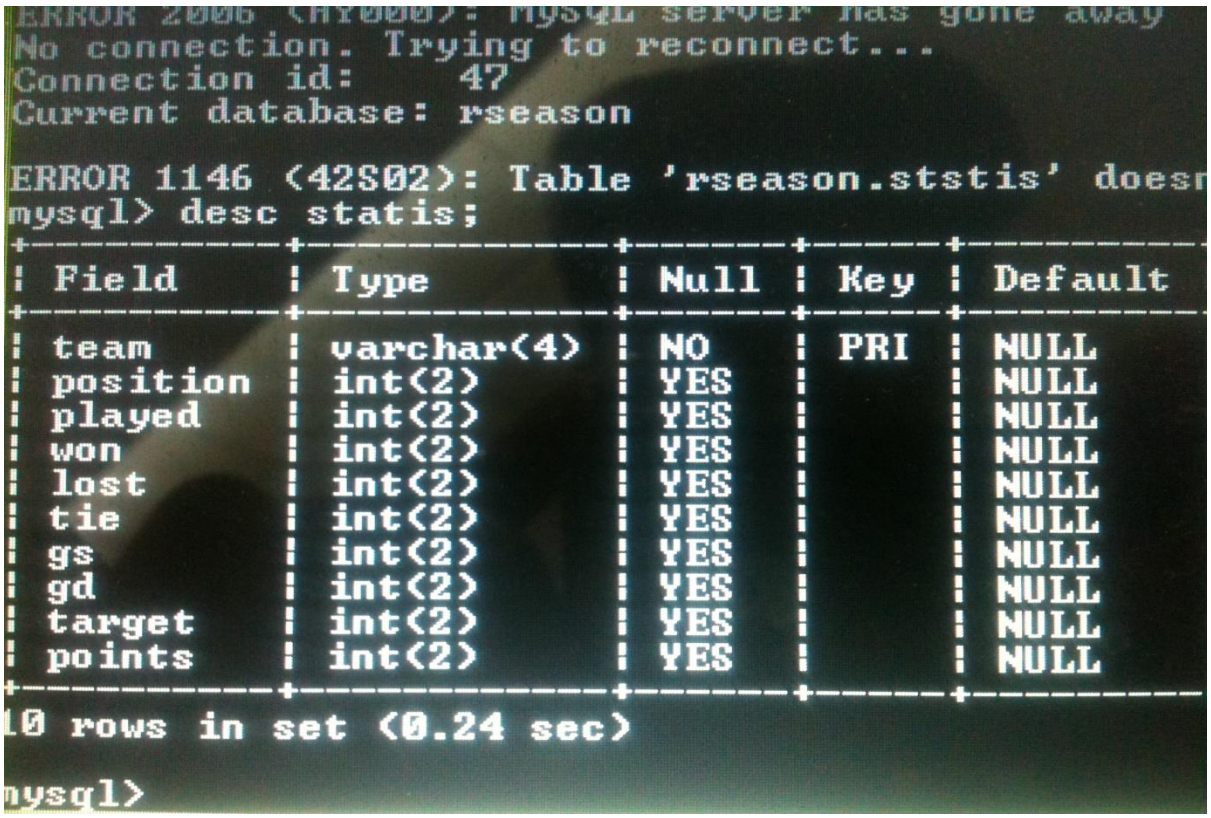
- **Names and numbers allotted to these tables in this section will be used throughout this document.**

Table name: “**statis**”

Table number: T.0.1

DESCRIPTION:

- The table **statis** stores all the data related to current season statistics.
- Current season means the running season in ENGLIGH PREMIER LEAGUE.
- With every match this table needs to be updated with the recent value for the attributes.



- This image shows us the description of the table **statis**.

- The **Field** section contains all the attributes* of the table.
- The **Type** section contains the type of the field attributes. That is, the type of an attribute, for example if it is of *character type, integer type* etc.

NOTE:

**ATTRIBUTE:* This is the factor that defines an entity.

***NOT NULL**: Any field marked with not null means that field must have some value that is it cannot be left empty.

***PRI:** This denotes the PRIMARY KEY. This means the attribute marked as primary key is unique for every other person. For example: password attribute while logging in is a primary key because no two people can have the same password.

• Attributes in STATIS table:

- Team: This contains the name of participating teams.
- Position: League table standing of the teams.
- Played: This is the total number of matches played by the team till date.
- Won: The number of matches the team has won.
- Lost: The number of matches the team has lost.
- Tie: The number of matches the team has tied.
- Gs: This is the number of goals scored by the team.
- Gd: This is the amount of goal difference.
- Target: this states the number of shots o target.

Table name: “auth”

Table number: T.0.2

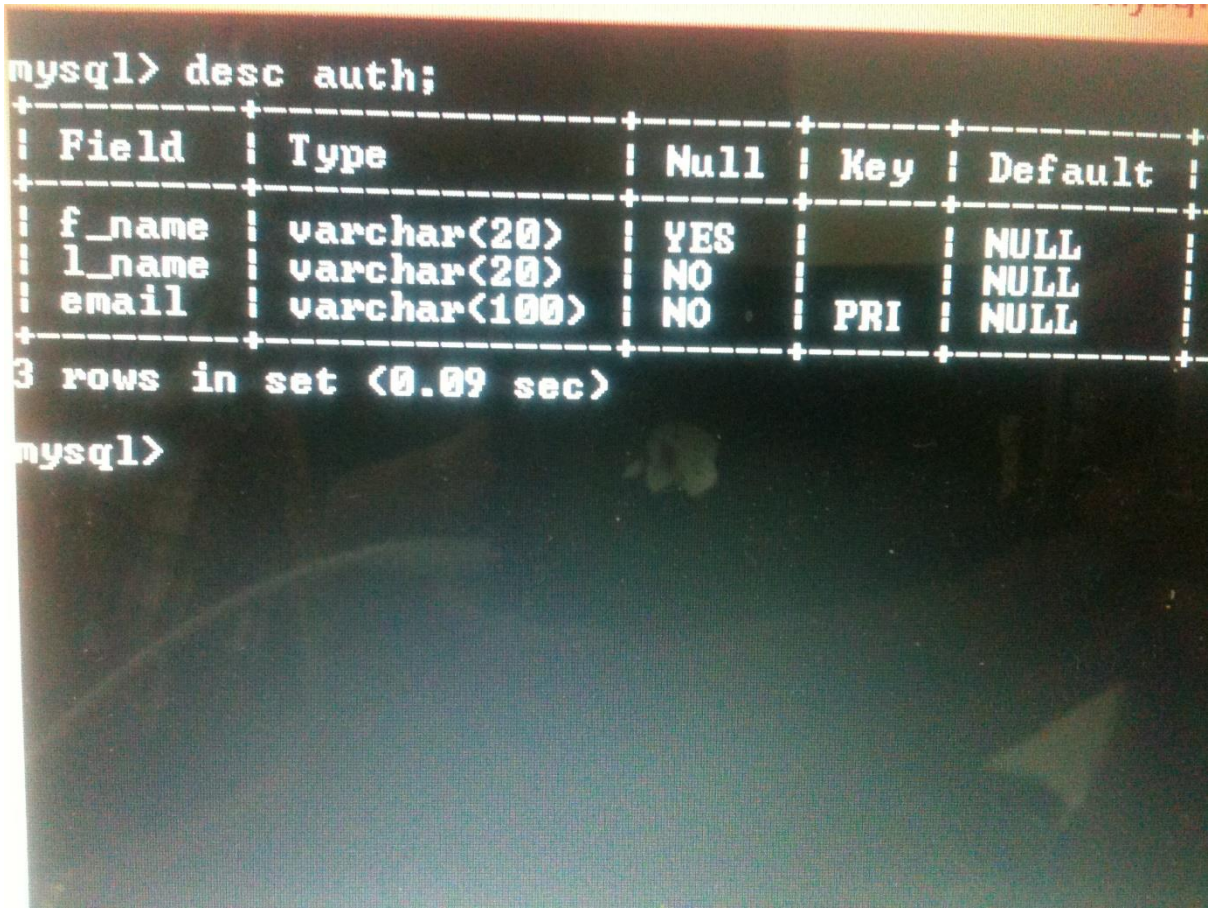
- **DESCRIPTION:**

  - This table consists of the login credentials.

  - In this table we have stored the email id, last name and first name of the authentic users.

  - This table is used as a validating table.

  - This table feeds value dynamically while signing up.



```
mysql> desc auth;
+---------+--------------+------+-----+---------+
| Field   | Type         | Null | Key | Default |
+---------+--------------+------+-----+---------+
| f_name  | varchar(20)  | YES  |     | NULL    |
| l_name  | varchar(20)  | NO   |     | NULL    |
| email   | varchar(100) | NO   | PRI | NULL    |
+---------+--------------+------+-----+---------+
3 rows in set (0.09 sec)

mysql>
```

•Attributes in the AUTH table:

- f_name: This field stores of the first name of the user.
- l_name: This field stores the last name of the user.
- email : This field stores the email of user.

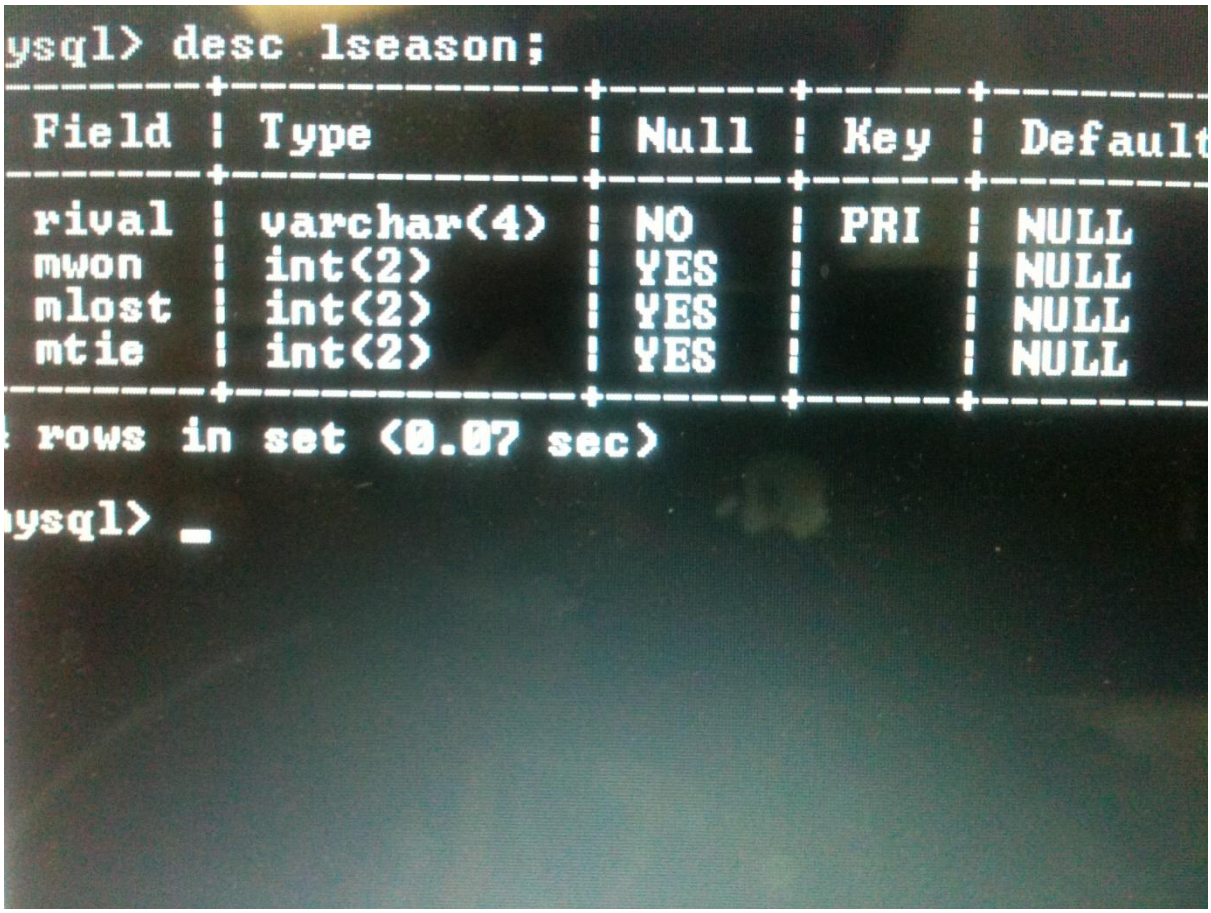
Table name: "lseaseon"

Table number: T.0.3

- **DESCRIPTION:**

  - This table is used to store the number of wins, defeat and tie for each team in the last 2 seasons.

  - This table will be used in the application to calculate the outcome of a football match.



•Attributes in the LSEASON table:

-**rival**: This field stores the name of the team.

-**mwon**: This field stores the number of matches won by the teams in last two seasons.

-**mlost**: This field stores the number of matches lost by the teams in last two seasons.

-**mtie**: This field stores the number of matches having a stalemate by the teams in last two seasons.

(I) Project Description

Project Overview:

SECTION: - 1.1

- India has a huge population and her population has a great

sympathy for football. In the last few decades **Indian Football Team** has improved to a great extent in world ranking. This shows the increasing rate of enthusiasm for football in India.

- Indians also has immense interest in club football. Mostly, people

watch **English Premier League; i**t's the world's most viewed and most famous league.

- Indians have been following this league for a long time now and

they have supporting some team. People follow these football leagues like addiction. Before, every match day the supporters look at the statistics of their team. They do this to get an idea about their supporting team against the rivals.

- It’s quite a troublesome work to browse different sites and find the statistics.

- This application has been designed to reduce the inconvenience of the fellow supporters.

WHAT ACTUALLY DOES THIS APPLICATION DO?

- This application actually gives the user every statistics about their supporting team and the rival team.

- It not only gives us the statistics, it analyses the stats and uses few machine learning algorithm to calculate the probability of outcome for both the teams.

- In this way it decreases the work and makes supporting our team more fun.

The Purpose of the Project:

SECTION: - 1.2

(1.2.1) Background of Project Effort:

- **(1.2.1.1) CONTENT:-**
  - This project is a fairly interesting project for the football enthusiasts.
  - This project will reduce the effort needed to predict a football game.
  - This project started as an Industrial Training project for my university.
  - The work was continued because of the constant effort of my mentor in TITAN Company.
  - Moreover, this project is a huge benefit for football enthusiasts because it will save them much time.

- **(1.2.1.2) <u>MOTIVATION</u>:-**

    - The motivation came from the urge to submit the project in college presentation.

    - The reduced struggle of football enthusiasts was also another motivating reason.

    - **<u>Mr.Venkatesh R, Senior IT Manager, TITAN COMPANY LIMITED</u>**, triggered the work of the project by keeping a track of the project in a regular basis.

- **(1.2.1.3) <u>CONSIDERATIONS</u>:-**

    - This project started after considering the serious problems faced by the football fans while searching for statistics about their supporting football team.

(1.2.2) Goals of The Project:

- **(1.2.2.1) CONTENT:-**

This project will decrease the amount of work done to browse all the data and statistics about the teams because all the data will be stored in the project database and the user will get any data he/she needs in one place. In addition to it this project also calculates the probability of outcome of any match using some effective machine learning algorithms.

- **(1.2.2.2) MOTIVATION:-**

  - **Mr.Venkatesh R, Senior IT Manager, TITAN**, triggered the work of the project by keeping a track of the project in a regular basis. He judged the algorithms used in this project and analysed the algorithms to make them more effective.

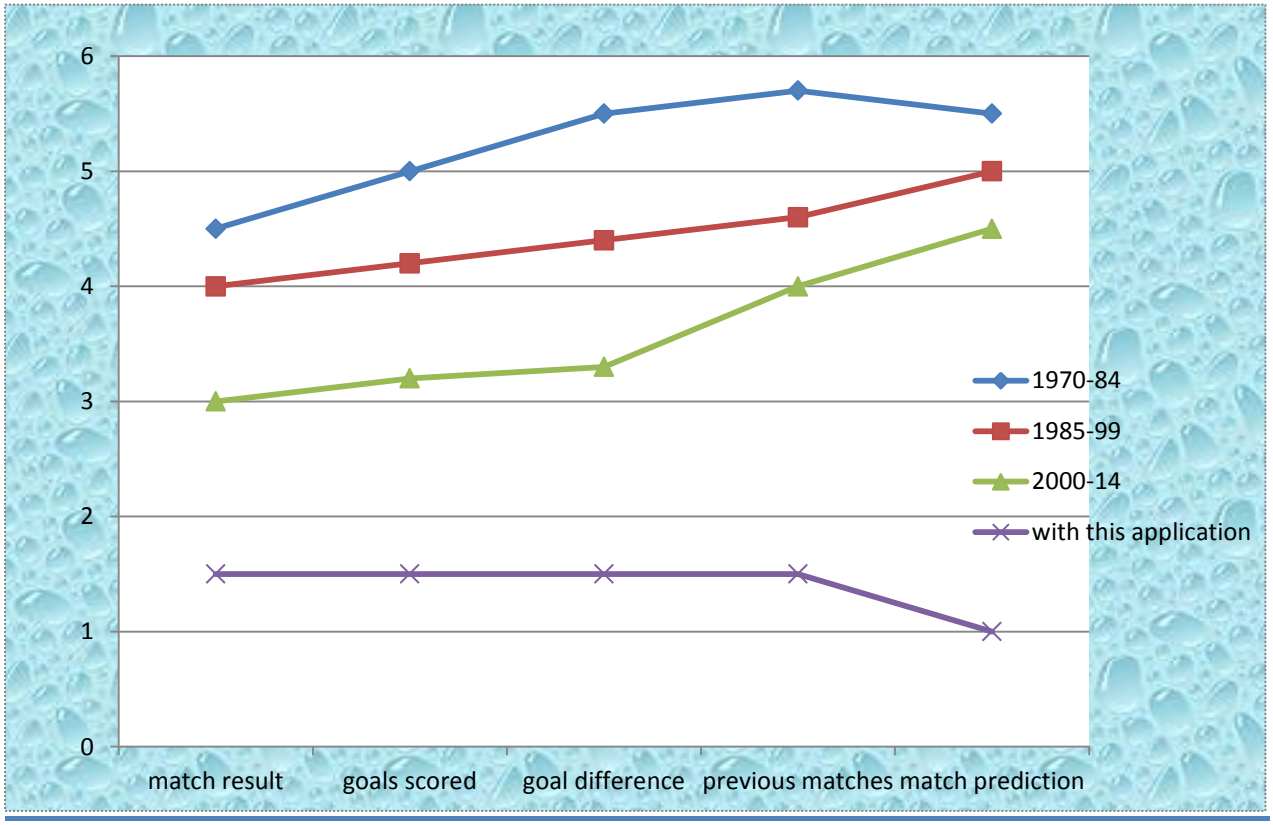
- **(1.2.2.3) EXAMPLES:-**

  - We want to give all relevant data and statistics about the team and the probability of the outcome as requested by the user.

(1.2.3) Measurement:

- This application will reduce the effort to find the track records of our favourite team and helps us to find a more efficient probability of match results.

- The graph below shows us the efficiency systems in finding such statistics with the advent of time.



- **GRAPH DESCRIPTION:**

  - **Along X-axis:**

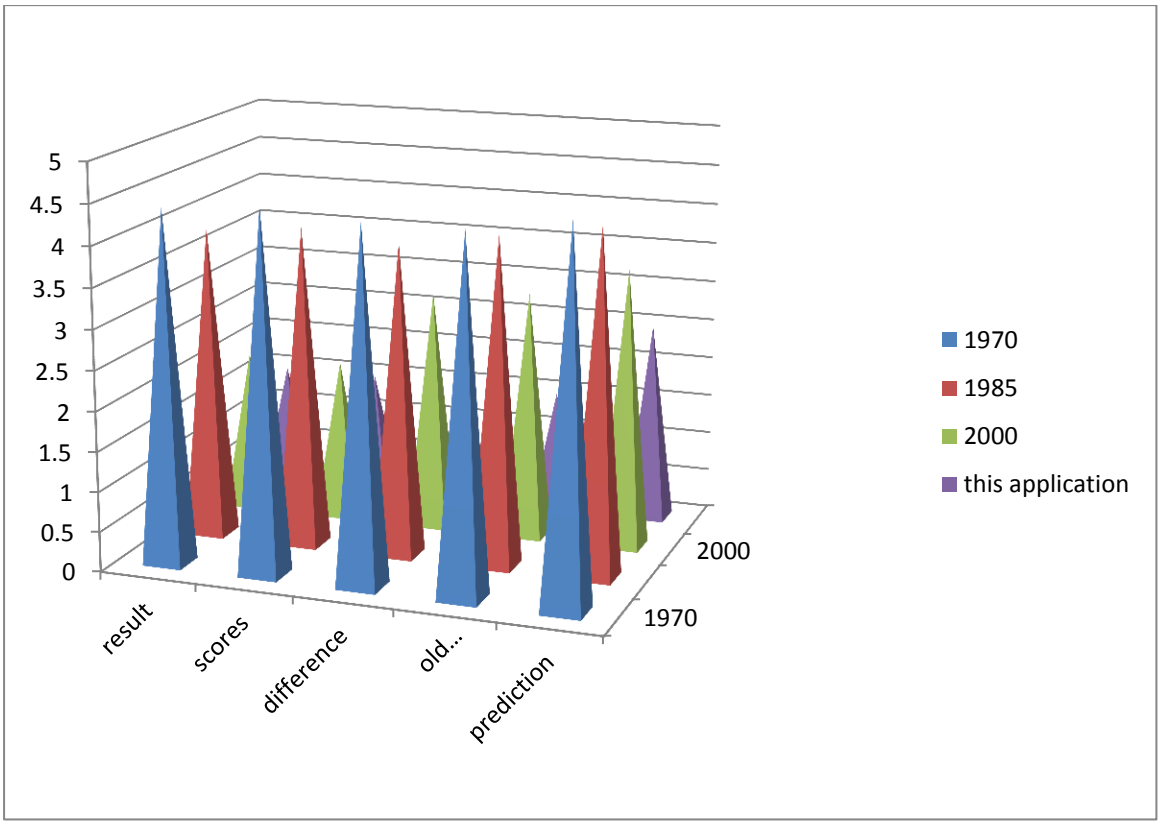
    This shows the *DIFFICULTY LEVEL* as a function of *EFFORT* and rated in a scale of **0** to **6.**

F(x) =f (effort)

- **Along Y-axis:**

This shows the factors that users need to predict a game outcome or these are the factors that generally users search before watching their favourite team's game.

- This graph shows the amount of effort required to search those values starting from 1970s to 2014 and Also shows the reduction in effort with the implementation of this application.



This graph shows us a clear idea about the effort we need to put for retrieving the value of those fields with the advent of years.
This graph also shows us the reduced effort due to the use of this application.

The Scope of the Project:

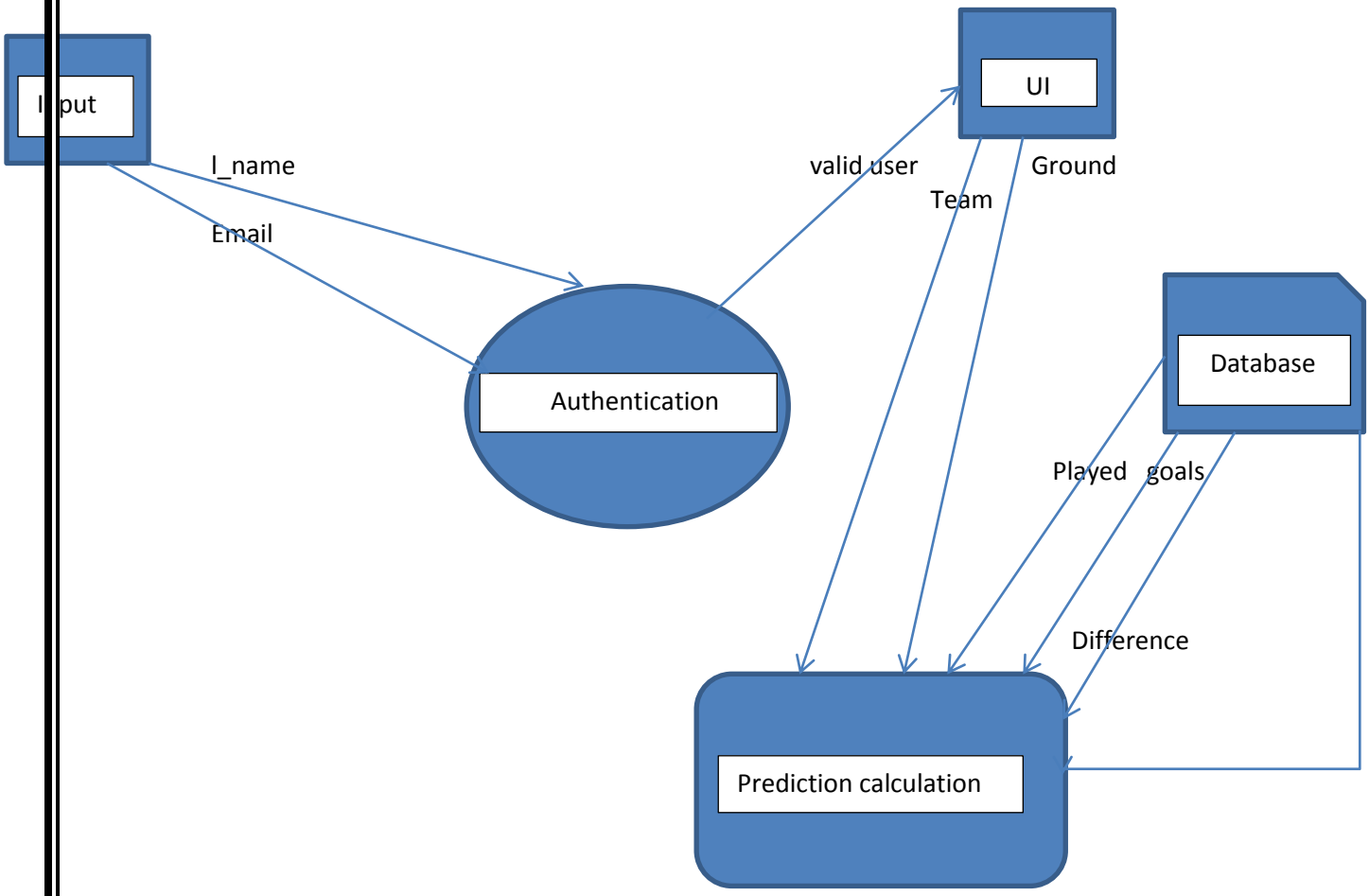
SECTION: - 1.3

(1.3.1) The Current Situation:

- **(1.3.1.1) CONTENT:-**
  - This project finds the probability of winning, losing and tie. To do this the project uses a high level algorithm for the calculation of the probability.
  - The algorithm uses the value of few old records. These records are maintained by a database used in this project.
  - The project also creates a display of the old records.
- **(1.3.1.2) MOTIVATION:-**
  - This project is for college use.

(1.3.2) The Context of the Work:

- **(1.3.1.1) CONTENT:-**
  - This project deals with databases for getting required information.
  - The authentication process deals with validation of authenticate users and thus restoring security purpose.
  - After authentication process is done the user interface part asks for the team name and the ground.
  - Based on this, the model finds required value of the probability.



Project Scenario:

SECTION: - 1.4

- The end users can use this product to get an idea about the old statistics which are very hardtop find.

- This product can also be used in detecting the winning, losing and tie probability of a particular game.

Maintenance and Services:

SECTION: - 1.5

- The project database will be constantly updated by the developers.

- The developers will update the database about the values of various attributes used to define the entity.

- Every time a football match ends the developers will update the value of the database leading to more effective database.

- The more recent values the database will contain the more efficient it will become.

- The part of the database that deals with the current season value needs to be updated regularly.

Stake Holders:

SECTION: - 1.6

- **SPONSOR:** TITAN CORPORATE.

- **MANAGER:** Mr VENKATESH R
  SENIOR MANAGER, IT

- **EXPERT:** Mr VENKATESH R
  SENIOR MANAGER, IT

- **DEVELOPER:** SURYADEEP CHATTERJEE
  STUDENT, COMPUTER SCIENCE

System Requirements:

SECTION: - 1.7

- This project will work in any system from windows XP to any latest version of operating system.

- System must have a minimum RAM of at least 128mb.

- System must have any one of the following browser;
  Google chrome, Internet explorer or Firefox.

(II) Requirements:

SECTION: - 2

Project Usecases:

SECTION: - 2.1

- The actors in this project that interact with the system are:

  - Users.
  - Authenticator
  - calculator

- These actors mentioned above play a part in the successful completion of the project.

- Each actor performs their individual work.

Usecase diagrams:

SECTION: - 2.2

- The usecase diagram is made using the software star UML.

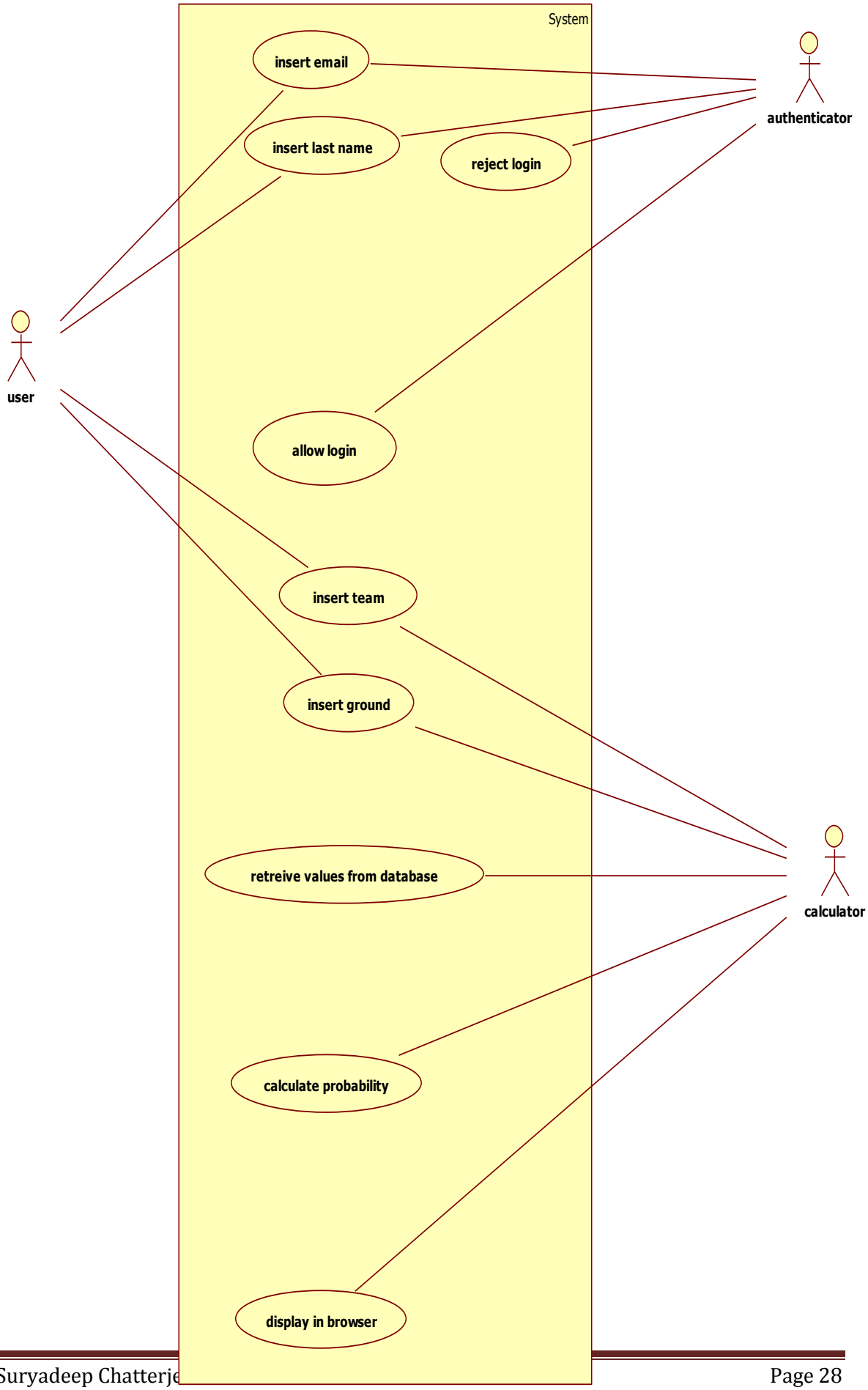
- The use case diagram is shown below.

- The AUTHENTICATOR checks the log in credentials filled by the user and finds out if it is valid or not.

- If the credentials are valid the user is allowed to enter the project page and fill the asked information there.

- After the user is done with the information feeding the calculator retrieves all the data required by it to find the required probability.

- Once the calculator is done with the calculation thing it displays the result in the browser.


System
insert email
insert last name
reject login
allow login
insert team
insert ground
retreive values from database
calculate probability
display in browser
authenticator
user
calculator

Data Requirement:

SECTION: - 2.3

- The project deals with various statistics to find the probability of the match outcome.

- The match outcome depends on aspects such as-

  # Number of matches played.

  # Number of matches won.

  # Number of matches lost.

  # Number of matches tied.

  # Number of goals scored.

  # Number of points scored.

  # Position of the team in table

  # Number of goal difference.

  # Number of shots in target

  # Number of matches won against the opponent in the last two seasons.

  # Number of matches lost against the opponent in the last two seasons.

- These values are stored in databases of the project.

- The database structure was shown in the beginning of this documentation in the section 0.

Project backend:

SECTION: - 2.4

- This is a web based project which is done using JAVA environment.

- The web pages are created using HTML tags.

- The web interactions are done using the JAVA servlet concepts.

- The database is maintained using the MySQL software.

- The values in database are retrieved using JDBC concept.

- The connection with the server is made using local host.

- Server used is APACHE TOMCAT.

Compatibility:

SECTION: - 2.5

- The project is compatible with any system with minimum requirements.

- This project is more of a calculating project, it has no such graphic arrangements.

- It's a very small sized project.

- Any system with Windows XP or more is eligible to use this project.

- The RAM requirement is very minimal.

- The system must have any one working web browser because it is a web based application.

Extensibility:

SECTION: - 2.6

- The project can be extended to any extent.

- The project can be updated with any new database.

- The database is globally uploaded thus any change in data can be associated with the relevant change in the required database.

- Thus any change in the real time data can be associate by changing the database.

Reliability:

SECTION: - 2.7

- The project has high value reliability.

- The project deals with data which has been tested by statisticians and updated in famous websites like espn.org/sports/epl.

- The algorithm used to find the value of probability is based on machine learning algorithm.

Security:

SECTION: - 2.8

- The security issues are well planned.

- The database can only be handled by admin thus there is no scope of data manipulation leading to ambiguous result.

- The users must be an authenticate user to use the application.

- The user can sign up with a valid email id thereby doing a proper security check.

Accessibility:

SECTION: - 2.9

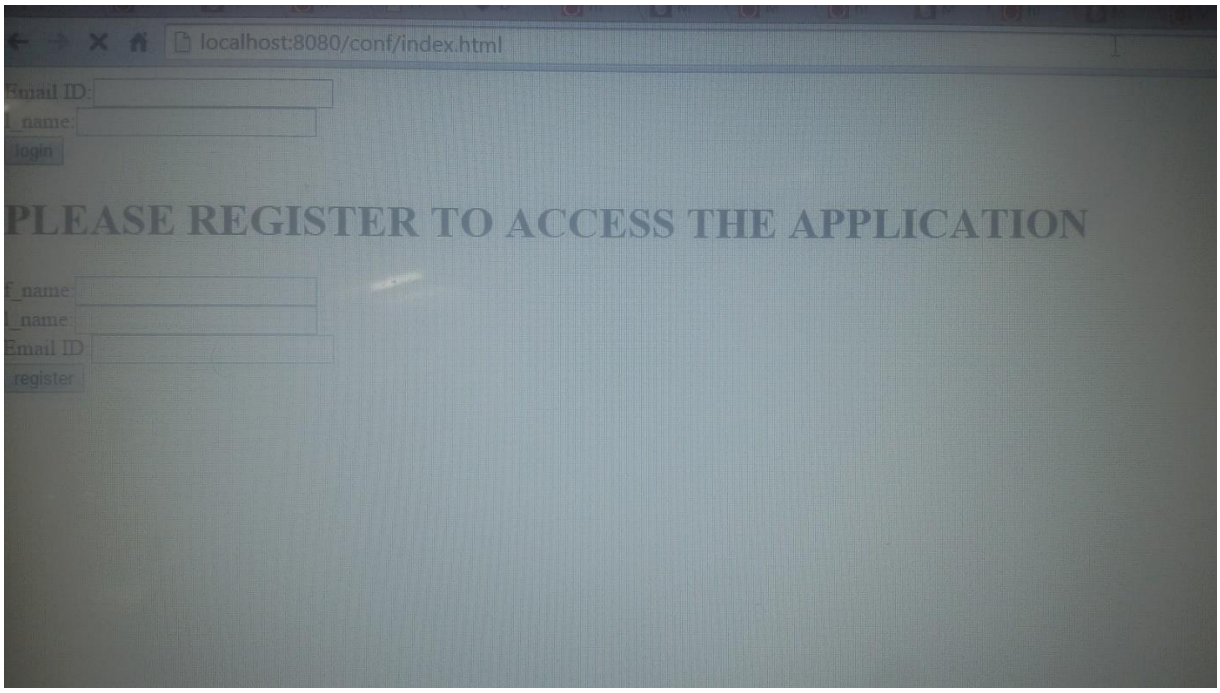
- In spite of having security login this application has an easy accessibility.

- Sign up process takes the users through three processes.

- It asks for first name, last name and a valid email id.

- Proper filling of these data creates a valid user account.

- This application is completely a web based application thus there will be no issue of downloading and installing.



- This image shows the log in page and the sign up page.

- The sign up is done in only 3 steps.

Longevity:

SECTION: - 2.10

- The application stands for various independent aspects.

- The application is a dynamic application, the values it requires is uploaded in a dynamic database.

- The dynamic database stores all the values required by the application to get valid results.

- The database is protected by admin rights and every time there has been any change is value the data is updated by the admin.

- The database table value has been mentioned in section 0 of this document.

Adaptability:

SECTION: - 2.11

- The application is adaptive to any system including any operating system as well as web browser.

- The application is also immune to changing database values.

- The database dynamically maintained by the admin.

(III) Design:

SECTION: - 3

▪ System Design:

SECTION: - 3.1

(3.1.1) Design Objective:

- The main objective of this project is to retrieve the values from the database and perform some calculations on these data and find the probability of the outcome.

- This is achieved by maintaining a database and making various classes keeping in mind the object oriented principles.

- These classes interact with the database and perform calculations.

(3.1.2) Class Architecture:

- The application has made use of normal JAVA classes as well as SERVLET class in JAVA.
- The list of classes and their description are given below:

Calc.java:

This class is used to compute the basic calculations with the values retrieved from the database. This is a normal java class.

Goaldiff.java:

This class is used to calculate the probability weightage caused due to the goal difference between the teams.

Goalscored.java:

This class is used to calculate the probability weightage caused due to the goal scored by the two teams.

ground.java:

This class is used to calculate the probability weightage caused due to the ground in which the match is being played.

Login.java:

This class is a JAVA servlet class that is used to get the values of login credentials in the browser page.

percen.java:

This java class converts the result into percentage format from the fraction format.

register.java:

This java SERVLET class is used to perform SIGN UP operation. This class is linked with a HTML page where the user needs to fill in the registering credentials.

season.java:

This java class is used to calculate the probability weightage caused due to the result in the last two seasons.

Targets.java:

This java class calculates the probability weightage due to the number of shots in target.

Teamvals.java:

This java class connects to the database and retrieves various values.

Validate.java:

This class check the login credentials and gives permission to the user to access the home page.

Values.java:

This java class stores all the values from the database.

Welcome.java:

This class welcomes the valid user and directs the user to a valid page for more information required for the calculation.

(3.1.3) Class Diagram:

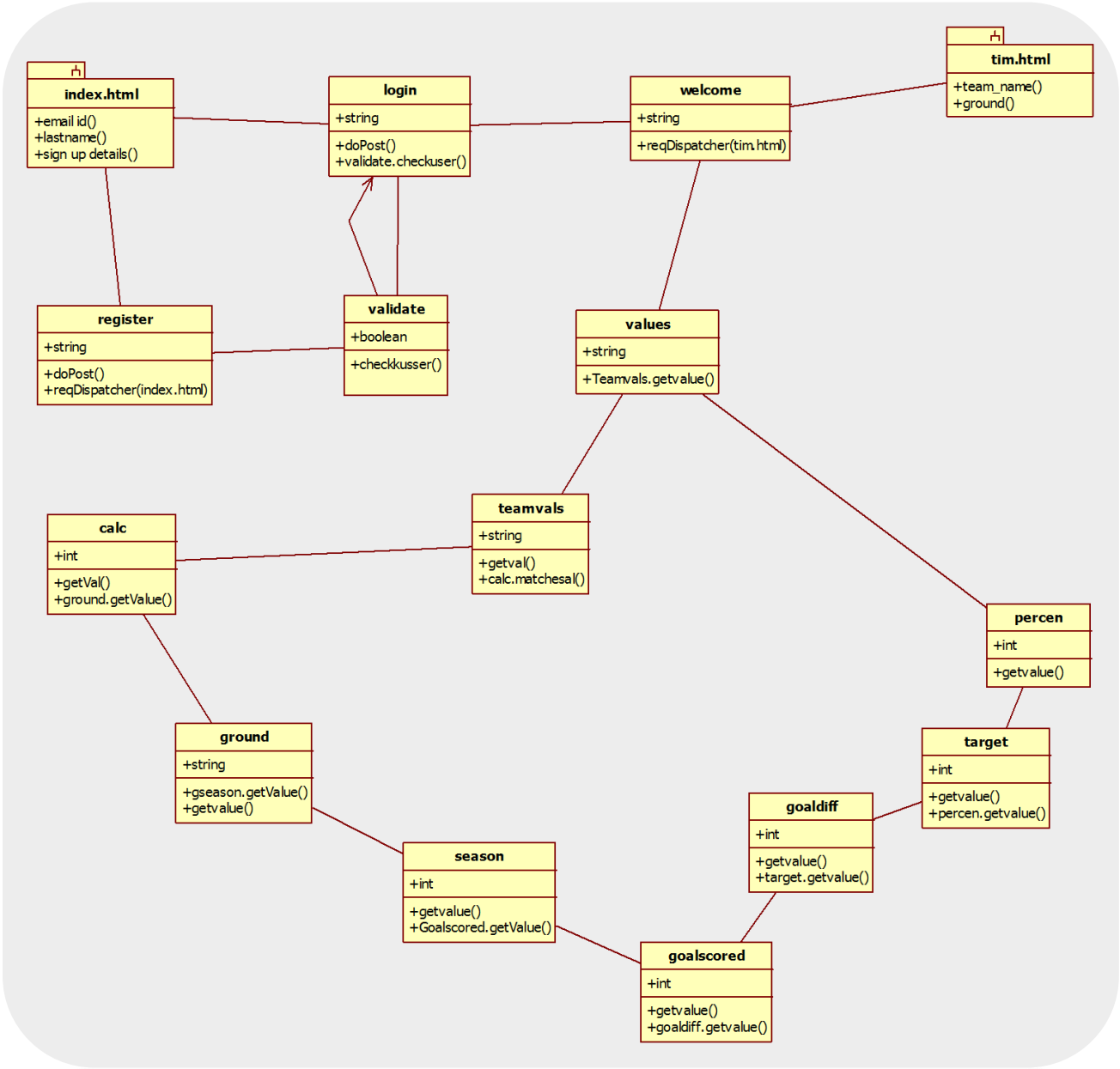
- The class diagram below shows the interaction between all the classes in the following application.

- The classes are also linked with html files in order to get values from user input.

- The 1st constraint in the boxes in the class diagram shows the class name.

- The 2nd constraint in the boxes in the class diagram is the data type the class deals with.

- The 3rd constraint in the boxes in the class diagram is the methods that the class deals with and interacts with.


index.html
+email id()
+lastname()
+sign up details()
login
+string
+doPost()
+validate.checkuser()
welcome
+string
+reqDispatcher(tim.html)
tim.html
+team_name()
+ground()
register
+string
+doPost()
+reqDispatcher(index.html)
validate
+boolean
+checkkusser()
values
+string
+Teamvals.getvalue()
teamvals
+string
+getval()
+calc.matchesal()
calc
+int
+getVal()
+ground.getValue()
percen
+int
+getvalue()
ground
+string
+gseason.getValue()
+getvalue()
target
+int
+getvalue()
+percen.getvalue()
goaldiff
+int
+getvalue()
+target.getvalue()
season
+int
+getvalue()
+Goalscored.getValue()
goalscored
+int
+getvalue()
+goaldiff.getvalue()

Software Architecture:

SECTION: - 3.2

- The software is implemented by categorizing it into three parts:

- The first part deals with html pages which creates user interface and takes values from user.

- The second part gets the value from the browser input by the user in the html page and stores it. This task is accomplished by the servlet java class.

- The third part deals with JDBC i.e. this part deals with the connection with the database server and retrieves the required value based on requirements and performs all the necessary calculations.

Project Calculations:

SECTION: - 3.3

To create the algorithm which can help us to predict a football game outcome, we need to decide the factors that affect the outcome of a football game. The factors are:

- Outcome of Last 'K' matches.
- Ground in which the match is played.
- Outcome of last seasons.
- Statistics of the team.
- Relegation zone.

(3.3.1) Last 'K' Matches:

- This section has a probability weightage of **0.225**.

- The team is allotted certain percentage of the total probability based on the outcome of the last 'K' matches.

- Outcome of last k matches mean the number of matches won, lost and tie by the team.

. Example:

Team A

Matches won: 3

Matches lost: 1

Matches tie: 2

Matches played: 6

According to algorithm:

Win probability:- ((3/6)*.225)

Lose probability:- ((1/6)*.225)

Tie probability:- ((2/6)*.225)

(3.3.2) Match Ground:

- The match ground plays an important role in outcome prediction of a football match.
- This section has a probability weightage of **0.225**.
- The home team gets a subtle advantage over the away team.
-

- Based on few statistics the calculation algorithm was designed.

. Example:

Team A:

Playing in home ground.

Team B:

Playing in away ground.

According to algorithm :

Team A:

Win probability:- (.65*.225)

Lose probability:- (.25*.225)

Tie probability:- ((.10*.225)

(3.3.3) Last Season:

- The weightage of this section is **0.225.**

- The last two seasons are kept in mind while finding the probability.

- The last encounters are given certain weightage based on the outcome of the matches.

. Example:

Team A in last two seasons against rival:

Matches won: 2

Matches lost: 0

Matches tie: 2

Matches played: 4

According to algorithm:

Win probability:- ((2/6)*.225)

Lose probability:- ((0/6)*.225)

Tie probability:- ((2/6)*.225)

(3.3.4) Statistics:

- This part has a total probability weightage of **0.225.**
- This part comprises of three parts:

~Goals Scored:

The number of goals scored by each team matters a huge in game prediction.

More number of goals scored by a team means the strikers are playing good thus there is a fair chance that the team will win.

This section carries 35% weightage of 0.225.

. Example:

Team A in last two seasons against rival:

Goals scored are 12 in 14 matches.

Team B in last two seasons against rival:

Goals scored are 18 in 14 matches.

According to algorithm:

Team A:
Win probability:- (.3*.0788)

Lose probability:- (.65*.0788)

Tie probability:- (.05*.0788)

~**Goal difference:**

The goal difference section carries a weightage of 45% of 0.225.

This part plays an important role in score prediction.

. Example:

Team A in last two seasons against rival:

Goal difference is 12 in 14 matches.

Team B in last two seasons against rival:

Goal difference is 18 in 14 matches.

According to algorithm:

Team A:
Win probability:- (.3*.10125)

Lose probability:- (.65*.10125)

Tie probability:- (.05*.10125)

Team B:
Win probability:- (.65*.10125)

Lose probability:- (.3*.10125)

Tie probability:- (.05*.10125)

~ **Shots on target:**

The shots on target section carries a weightage of 20% of 0.225.

This part plays an important role in score prediction.

. Example:

Team A in last two seasons against rival:

Number of shots on target is 12 in 14 matches.

Team B in last two seasons against rival:

Number of shots on target is 18 in 14 matches.

According to algorithm:

Team A:
Win probability:- (.3*.045)

Lose probability:- (.65*.045)

Tie probability:- (.05*.045)

Team B:

Win probability:- (.65*.045)

Lose probability:- (.3*.045)

Tie probability:- (.05*.045)

Software used:

SECTION: - 3.4

- The software used to do the java coding is ECLIPSE MARS.

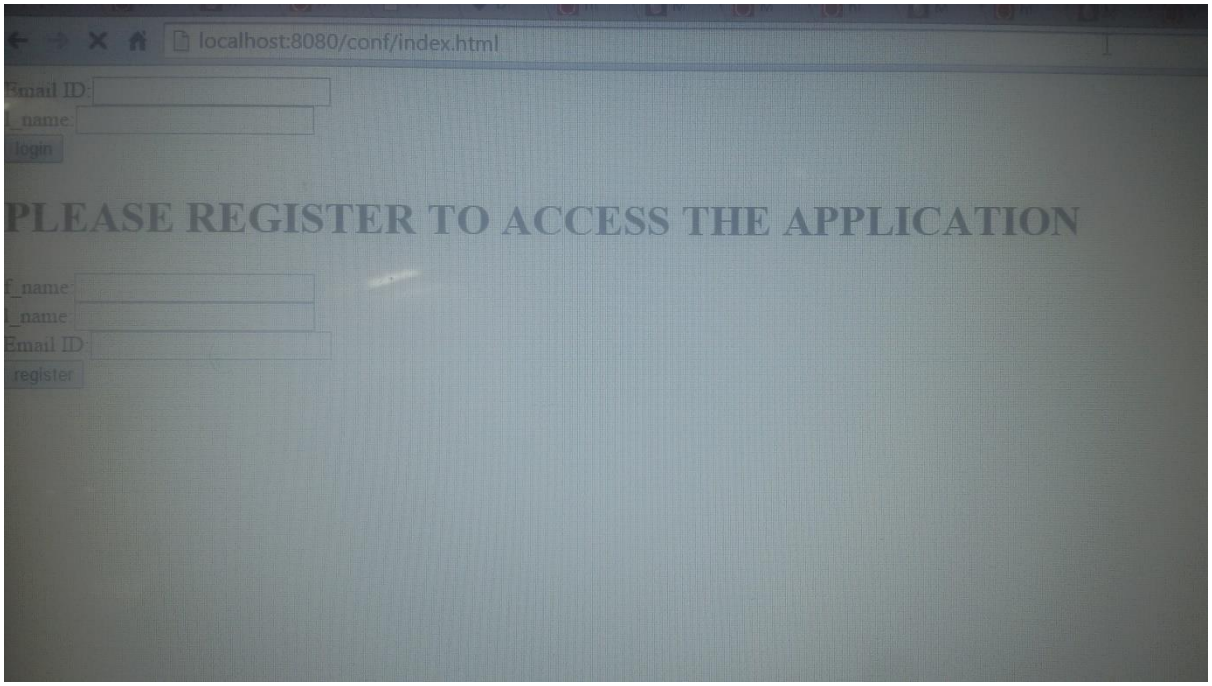
- The database is created using MySQL software.

- The eclipse is connected to the database server by database connector plugin.

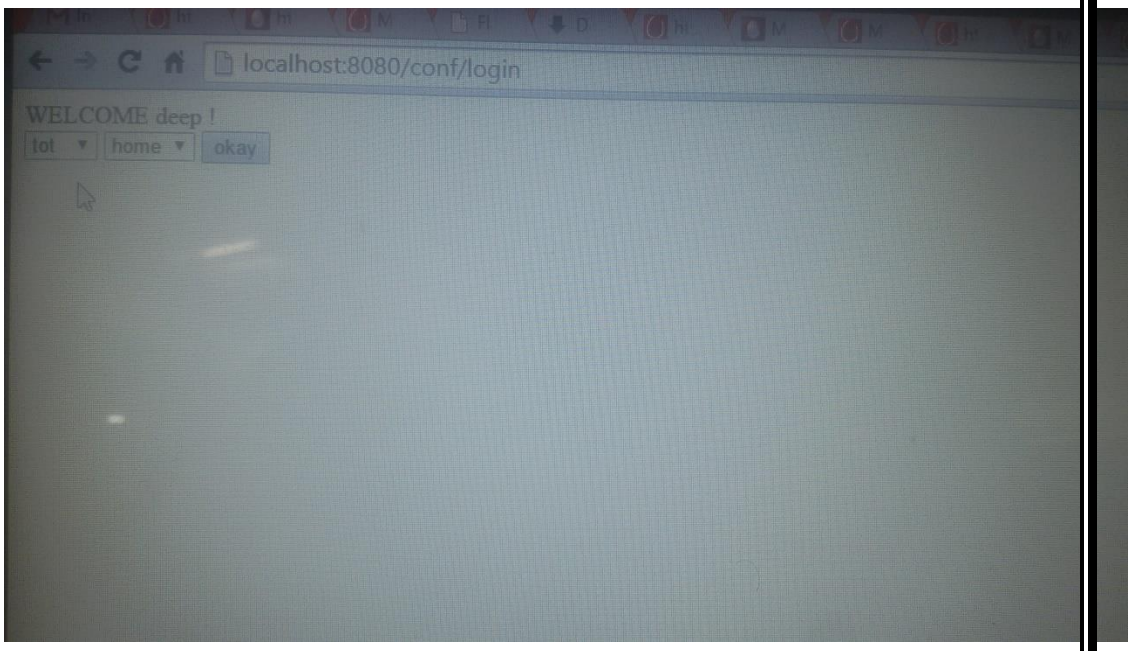
User Interface:

SECTION: - 3.5

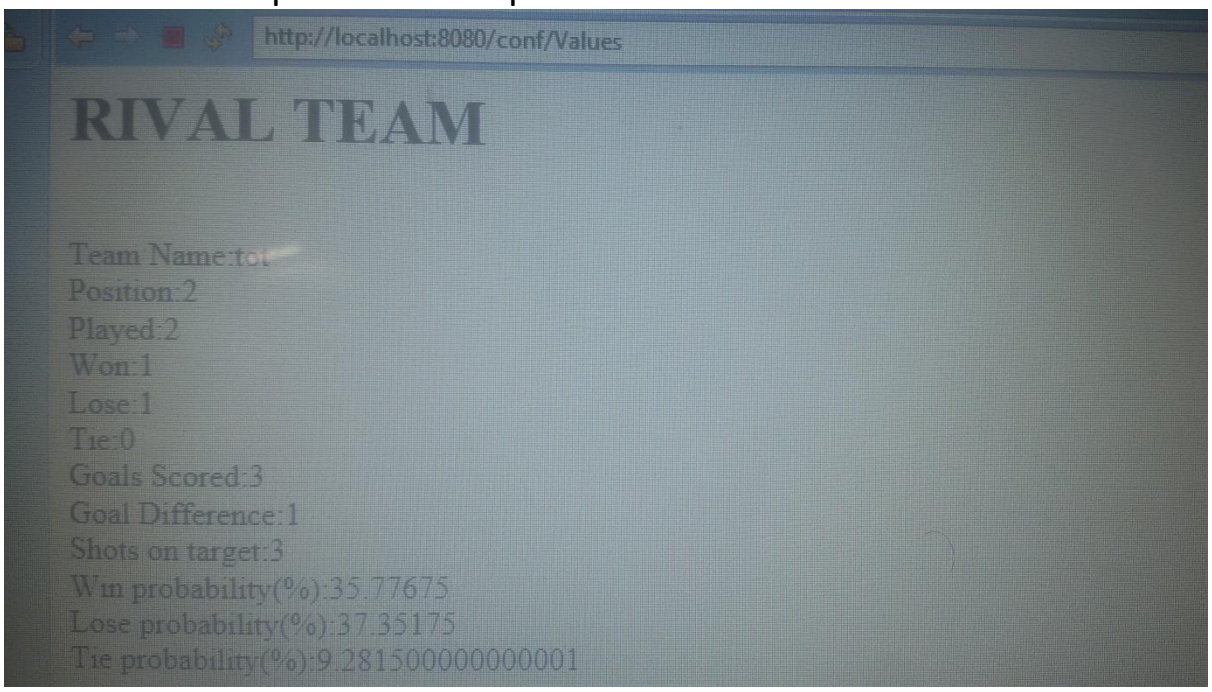
- At first the html page asks for log in credentials and if the user is not registered it asks the user to register.

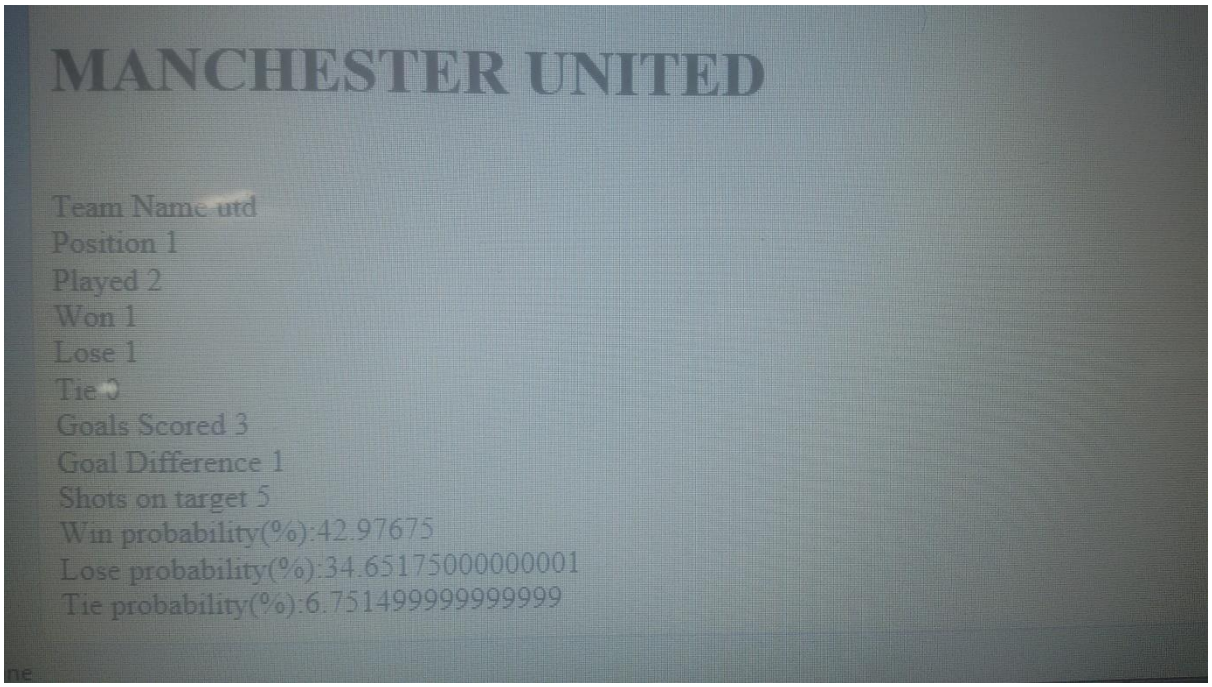


- After that another html page asks for rival team name and ground details.



- After entering the details class calculates everything and prints the output in the browser.





(IV) Test Plans

Section: 4

Test cases:

SECTION: - 4.1

To check the following application we designed a demo database for a small case and checked it. The test case is elaborated below.

(+) Demo Database:

- The database contains few set of values, including teams and team details.

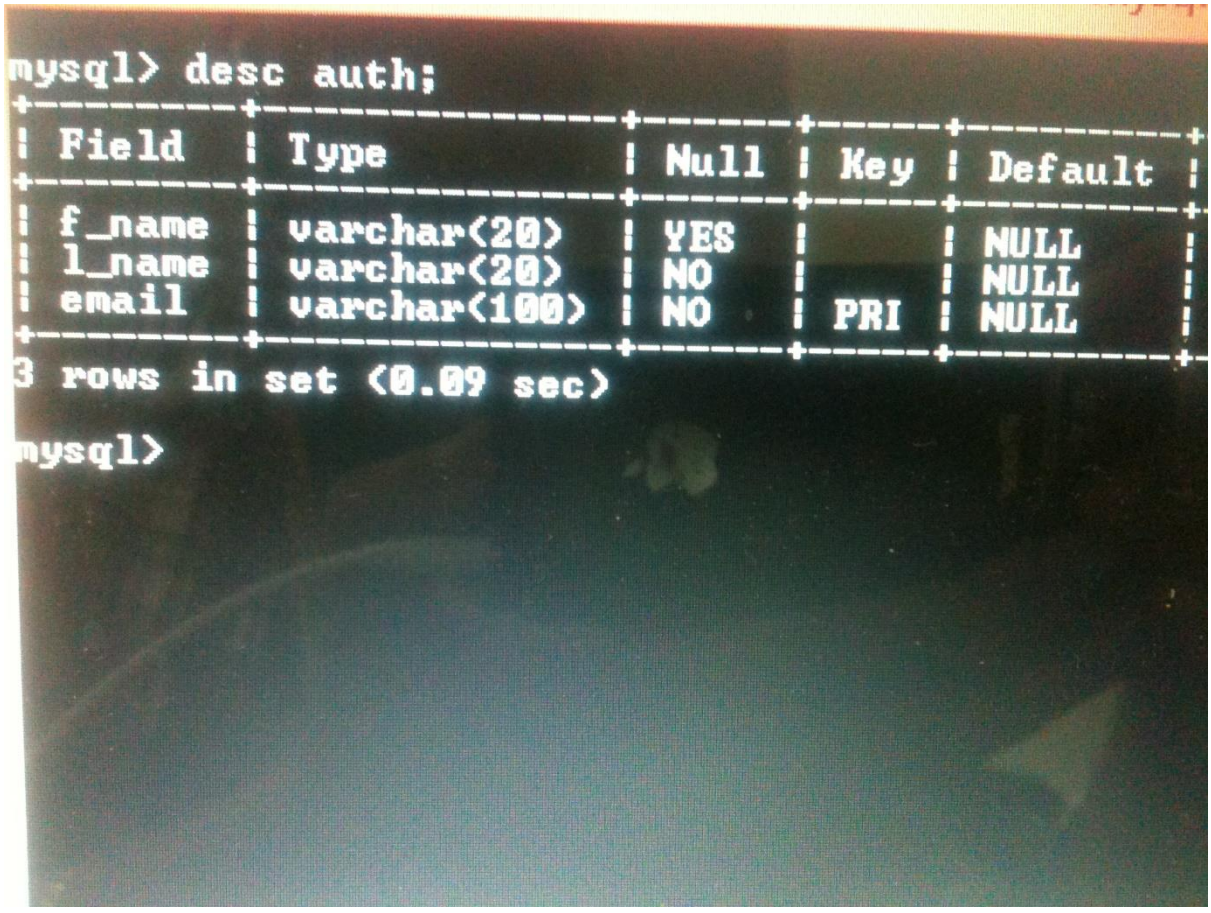
  (i) Database for authentication.
  (ii) Database for team details.
  (iii) Database for last two season details.

(4.1.1) Database for authentication:

- This database deals with the user authentication details.

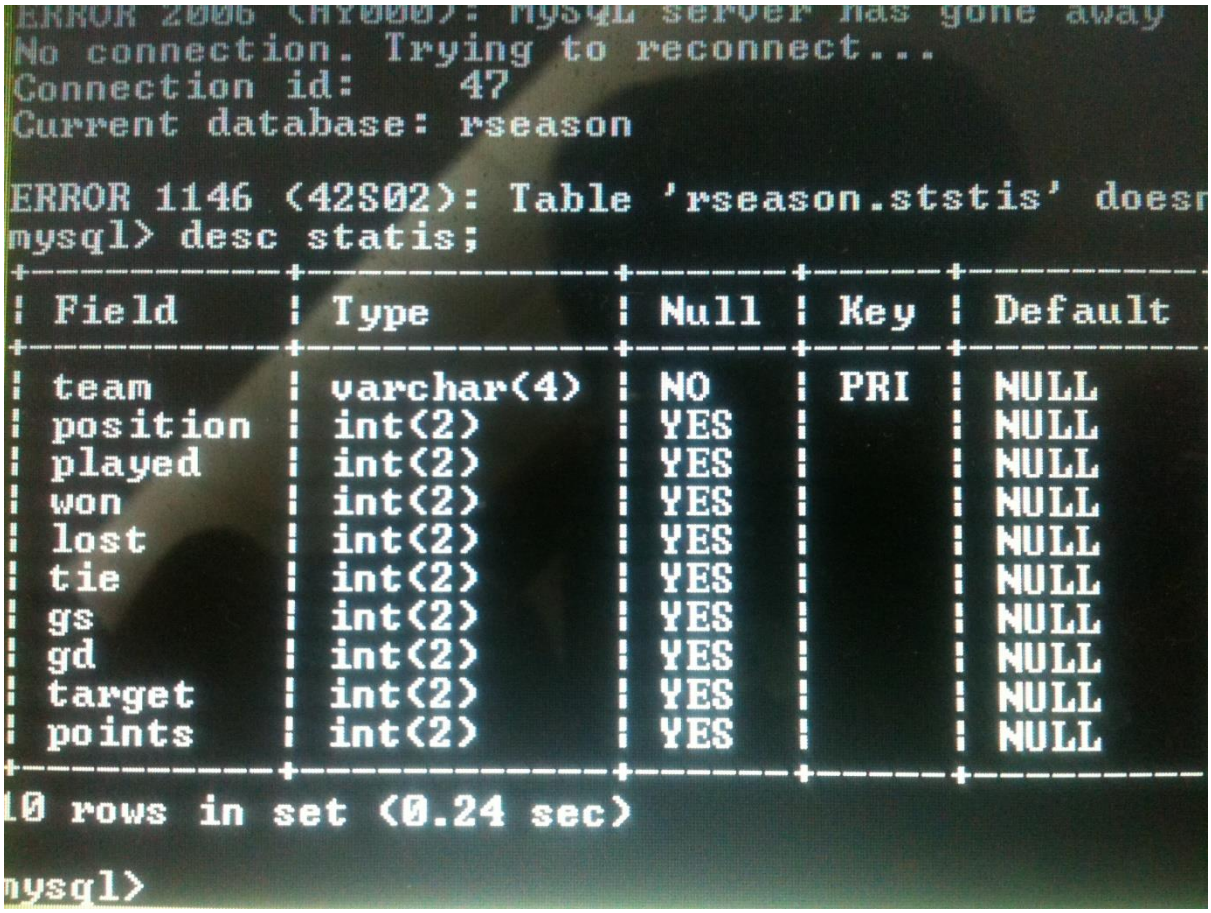
- When a new user registers the database updates with the provided information.

- If the user's log in fails the program throws an error.



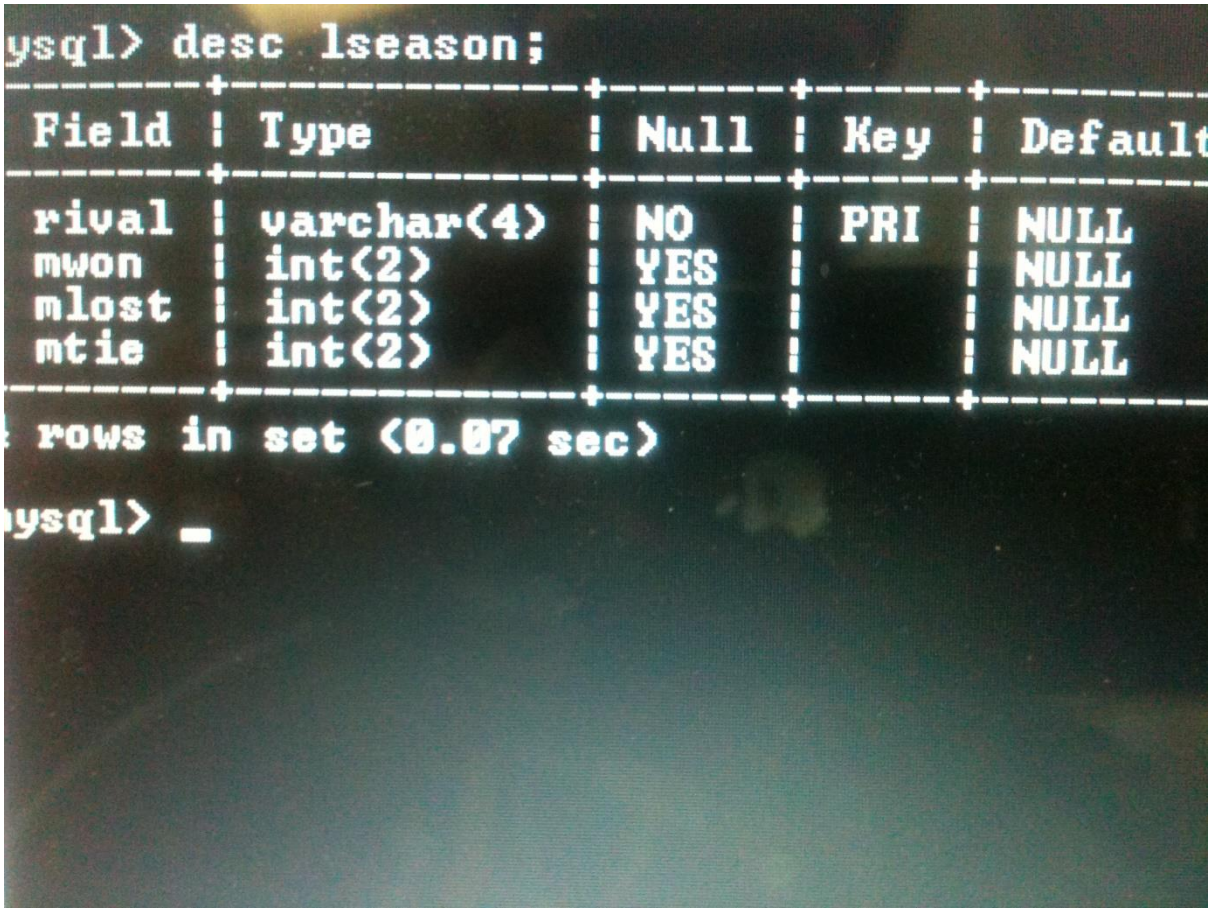
(4.1.1) Database for team details:

- This database stores all the relevant information about the participating teams.
- This database values are updated on a regular basis after the commencement of the season.

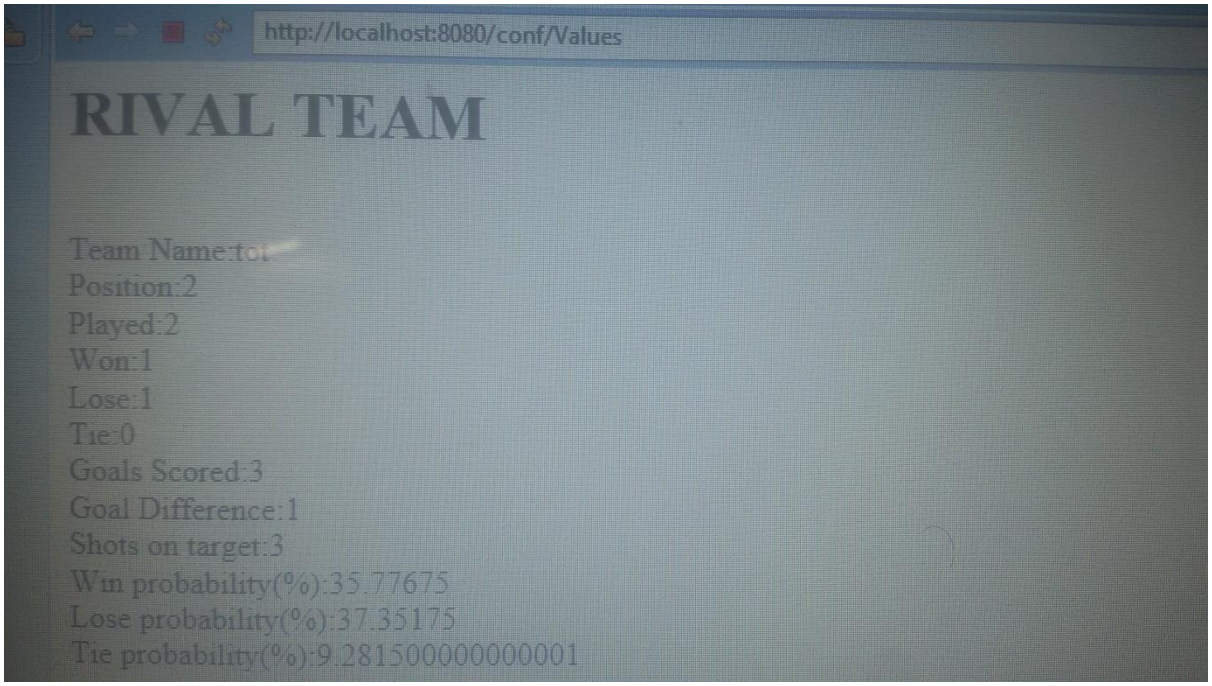


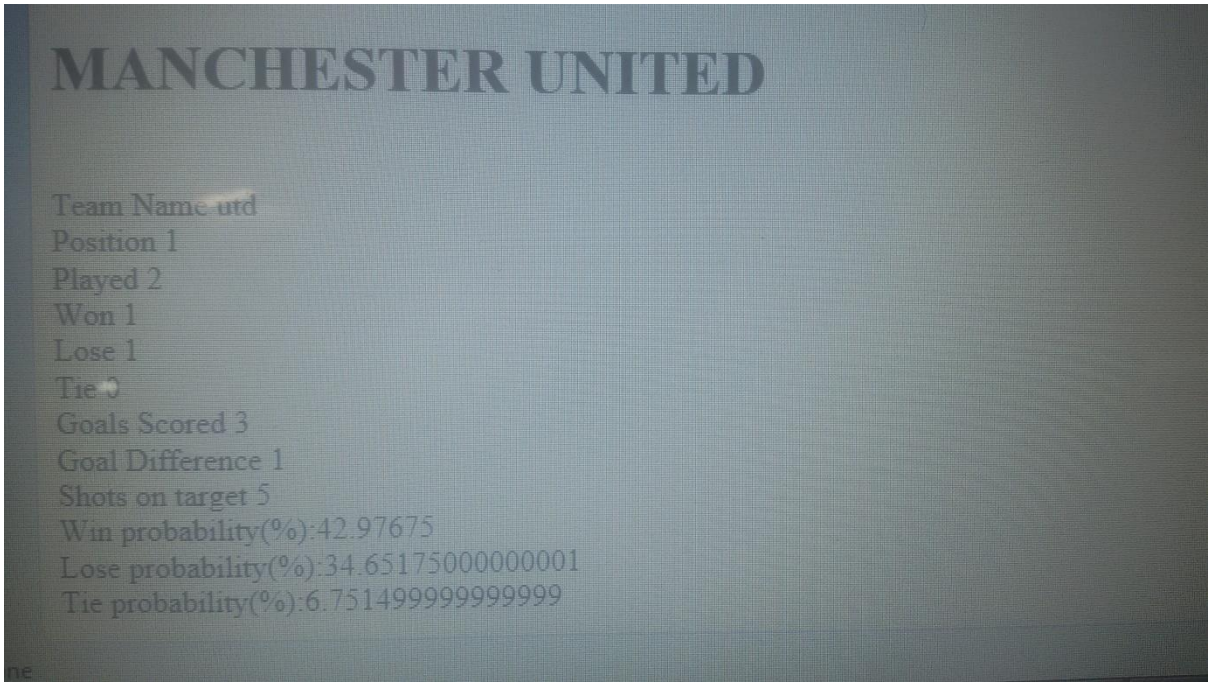
(4.1.3) Database for last season details:

- This database stores the values related to last season details.
- This database is used for calculation purpose.



(+) Output:


http://localhost:8080/conf/Values
RIVAL TEAM
Team Name:tot
Position:2
Played:2
Won:1
Lose:1
Tie:0
Goals Scored:3
Goal Difference:1
Shots on target:3
Win probability(%):35.77675
Lose probability(%):37.35175
Tie probability(%):9.281500000000001


MANCHESTER UNITED
Team Name utd
Position 1
Played 2
Won 1
Lose 1
Tie 0
Goals Scored 3
Goal Difference 1
Shots on target 5
Win probability(%):42.97675
Lose probability(%):34.65175000000001
Tie probability(%):6.751499999999999