**CRICKET MANAGEMENT SYSTEM**

By:

Karthik S (XII A9)

K S Shriram Kumar (XII A9)

**BONAFIDE CERTIFICATE**

Certified that this project is a bonafide work of

Master/Miss\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Roll No. of Class XII of **Maharishi Vidya**

**Mandir Sr. Sec. School**, Chetpet, Chennai during the year 2021

– 2022.

Date: 05/02/2022 Teacher – in – charge

Submitted for SSCE Practical Examination held in the **COMPUTER**

**SCIENCE** Laboratory at **Maharishi Vidya Mandir Sr. Sec.**

**School**, Chetpet, Chennai.

DATE : INTERNAL EXAMINER

PRINCIPAL EXTERNAL EXAMINER

SEAL

**ACKNOWLEDGEMENT**

I express my sincere gratitude to our  **PRINCIPAL, SHRI. G.**

**HARIBABU** and **VICE PRINCIPAL**,  **Smt V.SUNDARI** of our

s

institution for their continuous  support and encouragement.

I express my sincere thanks to my  Computer Science

Teachers, **Smt. T. SUJATHA** and **Smt. S. KRISHNAPRIYA** for

helping me to complete this project successfully.

I would also like to thank our Laboratory Assistant **Smt. A.**

**VIJAYA** for all the extended help to us

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **SNO** | **TOPIC** | **PAGE NUMBER** |
| 1 | Introduction | 5 |
| 2 | Database management system | 6 |
| 3 | Problem Definition | 7 |
| 4 | Problem Analysis | 10 |
| 5 | Hardware and software requirement | 13 |
| 6 | Source Code | 14 |
| 7 | Related Tables and Output | 28 |
| 8 | Project Impact | 36 |
| 9 | Future Scope and Development | 37 |
| 10 | Bibliography | 38 |

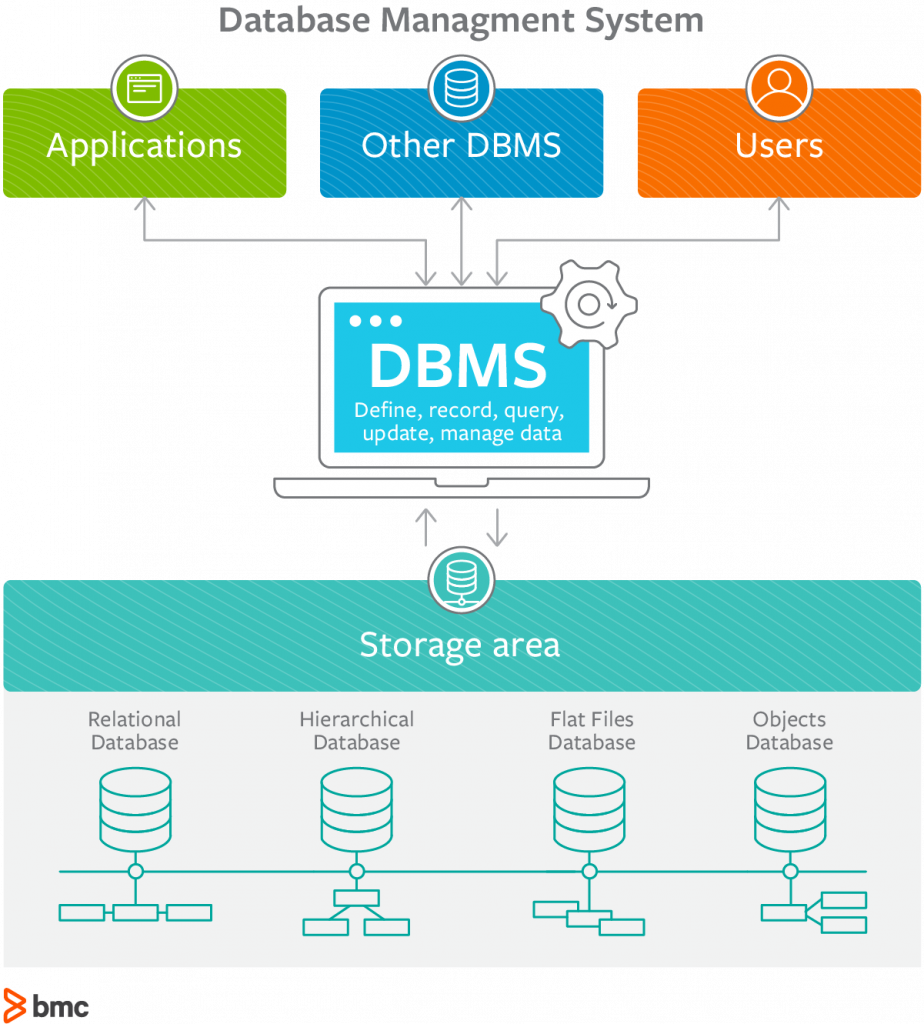
**INTRODUCTION**

India is a country where cricket is the most worshiped sport. Cricket in our country isn’t just another sport but a religion. A religion that people devote so much of their time and energy to, that it has paved way for different formats of the game like test matches, one day international and Twenty20s. Due to the sport’s immense popularity, this arena has a lot of technological and analytical opportunities. The Indian Premier League (IPL) is a professional Twenty20 cricket league that has grabbed the attention of audience all over the globe. Right from choosing players during auction to predicting teams in fantasy leagues, board members and cricket lovers have always sought to leverage database management systems and streamline many processes since the advent of technology.

It is this arena that we intend to capture via this project and build a database management system (DBMS) to better manage and understand the abundance of data available for cricket players while it comes to auctions, selections, and fantasy games.

**DATABASE MANAGEMENT SYSTEM**

DBMS stands for "Database Management System." In short, a DBMS is a database program. Technically speaking, it is a software system that uses a standard method of cataloging, retrieving, and running queries on data. The DBMS manages incoming data, organizes it, and provides ways for the data to be modified or extracted by users or other programs.



Some DBMS examples include MySQL, PostgreSQL, Microsoft Access, SQL Server, FileMaker, Oracle, RDBMS, dBASE, Clipper, and FoxPro. Since there are so many database management systems available, it is important for there to be a way for them to communicate with each other. For this reason, most database software comes with an Open Database Connectivity (ODBC) driver that allows the database to integrate with other databases. For example, common SQL statements such as SELECT and INSERT are translated from a program's proprietary syntax into a syntax other databases can understand.

**PROBLEM DEFINITION**

The aim of this project is to provide complete end-to-end information about cricket players like their batting/bowling averages, match history, experience and achievements along with their personal details. *A database management system is created with tables with player specific fields and attributes like runs scored, wickets taken, strike rates, economy, nationality and career-best figures*. Using SQL, we query and arrive at our desired result through the following process:

* The user is taken through a menu-driven program where he/she has to pick specifics that might interest them in a good player
* After taking all of the user’s choices (mentioned previously) into consideration, a list of most suited national or international players is shortlisted and the user is notified
* The user can also take a look at the statistics and performance of any player in the pool and can then base his decision to whether or not pick that player for his use case

This project intends to help auctioneers make quick decisions. It also aids fantasy game audiences to choose the best player for their team and accumulate points by assessing the player’s profile, which further translates into rewards.

“ Cricket database- a place to
find all the players career info“
4
View profile
info
Remove a
player profile
Add new profi...

**PLAYER TYPE:**

**BATSMAN STATISTICS (PROFILE INFORMATION)**

Age, Nationality, Current team

Matches played

Preferred Batting Position

Runs scored (T20)

Batting Average

Strike Rate

30+,50+,100+ scores

**PLAYER TYPE:**

**BOWLER STATISTICS (PROFILE INFORMATION)**

Age, Nationality, Current team

Matches played

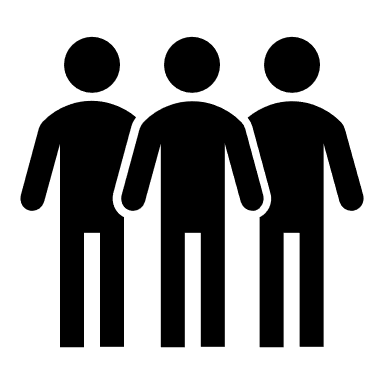
Wickets Taken (T20)

Bowling Average

Economy

3fer,5fer Wicket hauls

Personal Best figures

**OTHER INFORMATION**

Last auction price

Fantasy points

Number of seasons played

Number of teams played

**PROBLEM ANALYSIS**

All the necessary modules and libraries including the MySQL connector.

The landing page is a menu that requires the end user to specify if he is the owner of an IPL team, a regular fantasy game player or an admin.

* 1- A cricket player
* 2- IPL Team Owner
* 3- Admin

Every user requires a username and password for access. For a first time user, username and password is created by the admin. The user enters 1,2 or 3 and this choice is inputed.

**1-Cricket Player**

In case of option 1(Cricket Player), the user can enter his name and the following general stats of that particular player are displayed using a search algorithm.

* Player Name
* Nationality
* Age
* Current team
* Specialty (Batsman, Bowler, Allrounder or Wicketkeeper)
* Matches played
* Auction Price

**2-IPL Team Owner**

If the user chooses option 2(team owner), then he/she is taken to a page with the list of options namely

* 1- Viewing statistics of a particular player
* 2- Choosing player(s) with specified conditions

If the user chooses the first option from the above list, then he is expected to enter the player name of his choice and all the corresponding statistics of the player is displayed

If the user chooses the second option from the above list the user has to specify the statistics of a player that interests them. A brief questionnaire is shown to users, and they fill in the required information.

* Age
* Player Type (Batsman/Bowler/All Rounder)

|  |  |  |
| --- | --- | --- |
| **Batsman** | **Bowler** | **All Rounder** |
| * Batting Average * Strike Rate * Batting position | * Bowling Average * Economy * Bowler Type (Pace or Spin) | * Batting Average * Strike Rate * Batting position * Bowling Average * Economy * Bowler Type |

Once the users enter their choices for the above queries, an implicit +/- 10 units is assigned to all the parameters. For instance, if someone enters strike rate as 100, then an auto +/- range is assigned as it’ll be tough to have an accurate match. This would present a detailed list of potential batsmen/bowler or all rounder and the user can then leverage it for his/her use case- be it to form his/her fantasy league team or choosing a player to buy during auction.

9
public
Bowler
Batsman
Fielder
Keeper
Who are they
Average
Economy
Batting style
Role
 

**3-Admin**

If the user chooses option 3(Admin), then he is directed to a window where he chooses to:

1. Add a new player(s) information
2. Delete an existing player’s information
3. Modify an existing information

The above tasks are written as separate functions which are invoked as per the user’s command.

**HARDWARE AND SOFTWARE REQUIREMENTS**

|  |  |
| --- | --- |
| OS | UNIX Ubuntu 16.04 LTS or any OS which supports Python v 3.6.3 |
| SYSTEM TYPE | X86-64bit based processor |
| PROCESSOR | Intel Pentium [CPUN3540@2.16GHz](mailto:CPUN3540@2.16GHz) x 4 |
| CPU | 2.4 GHz |
| MEMORY | 8 GB |
| FRONT END | Python v 3.6.3 |
| BACK END | MySQL v8.0 |

**SOURCE CODE**

|  |  |  |  |
| --- | --- | --- | --- |
|  | from Player\_module import Player  from TeamManager\_module import password,Team\_manager  from Admin\_module import Admin  import mysql.connector as sql  connect = sql.connect(host= "localhost",user= "root",passwd = "shriram123",database = "sample")  cursor = connect.cursor()  print("\t\t\t\t\tWELCOME TO PLAYER SEARCH")  print('''Do you want to login as a:  1.Player  2.Team Manager  3.Admin''')  while True:      USER = input("Type your Choice:")      if USER == "Player":          Player()          break      elif USER == "Team Manager":          Team\_manager()          break      elif USER == "Admin":          Admin()          break      else:          print("Please enter a given choice your choice was invalid")  import password  def Team\_manager():      import mysql.connector as sql      import Player\_search      connect = sql.connect(host="localhost", user="root", passwd="shriram123", database="sample")      cursor = connect.cursor()      password.password('team manager')      while True:          print('''Do you want to          1.Search for a specific player          2.Use stats to pick group of players''')          search = int(input("Enter '1'for choice 1 and '2' for choice 2:"))          if search == 1:              name = input("Enter a player name:")              st = "select speciality from players where playername='%s'"%(name,)              cursor.execute(st)              speciality = cursor.fetchone()              speciality = speciality[0]              if speciality == "AR":                  st = '''select \* from players,batsmen,bowlers                               where players.playerid = batsmen.playerid                               and batsmen.playerid = bowlers.playerid and playername = "%s"'''%(name,)                  print('playername', 'nationality', 'age', 'current\_team', 'speciality', 'no\_matches', 'Price'                        , sep='|', end="|")                  print('avg\_runs', 'strikerate', 'preffered\_position', 'bestcore', 'Runs', '', sep='|', end="")                  print("economy", "wicket taken", "bowling avg", 'bestfigure', 'Type', sep='|')                  cursor.execute(st)                  for i in cursor:                      for j in i[1:12]:                          print(j, end="|")                      for k in i[13:14]:                          print(k, end="|")                      for l in i[15:20]:                          print(l, end="|")                      print()                  break              elif speciality == "batsmen" or "WK":                  st = "select \* from players,batsmen where players.playerid = batsmen.playerid and  playername ='%s'"%(name,)                  print('playername', 'nationality', 'age', 'current\_team', 'speciality', 'no\_matches', "price"                        , sep='|', end="")                  print('avg\_runs', 'strikerate', 'preffered\_position', 'bestcore', 'runs', sep='|')                  cursor.execute(st)                  for i in cursor:                      for j in i[1:12]:                          print(j, end="|")                      for k in i[13:14]:                          print(k, end="|")                      print("\b")                  break              else:                  st = "select \* from bowlers,players where players.playerid = bowlers.playerid and playername  = '%s'"%(name,)                  print('playername', 'nationality', 'age', 'current\_team', 'speciality', 'no\_matches', "price"                        , sep='|', end="")                  print('avg\_runs', 'strikerate', 'preffered\_position', 'bestcore', 'runs', sep='|')                  cursor.execute(st)                  for i in cursor:                      for j in i[1:12]:                          print(j, end="|")                      for k in i[13:14]:                          print(k, end="|")                      print("\b")                      break          if search == 2:             Player\_search.player\_search()             break  def Admin():      import password      import mysql.connector as sql      connect = sql.connect(host="localhost", user="root", passwd="shriram123", database="sample")      cursor = connect.cursor()      password.password('Admin')      print('''Do you want to:      1.Add records      2.Modify records      3.Delete records''')      while True:          Action = input("Enter 'Add' or 'Modify' or 'Delete' to perform respective function:")          if Action == "Add":              while True:                  import mysql.connector as sql                  import Player\_search                  connect = sql.connect(host="localhost", user="root", passwd="shriram123",  database="sample")                  cursor = connect.cursor()                  player\_id = input("Enter character id:")                  player\_name = input("Enter name of player:")                  age = int(input("Enter age:"))                  current\_team = input("Enter current team:")                  nationality = input("Enter nationality:")                  speciality = input("Enter speciality:")                  no\_matches = int(input("Enter no matches played:"))                  password = input("Enter password for player:")                  price = int(input("Enter Price:"))                  stgeneral = 'insert into passwrd values("{0}","{1}","{2}","Player")'.format(player\_id,player\_name,  password)                  cursor.execute(stgeneral)                  if speciality == "Bowler":                      economy = int(input("Enter economy:"))                      wickets\_taken = int(input("Enter wickets taken:"))                      bowling\_avg = int(input("Enter bowling avg:"))                      bestfigure = int(input("Enter best figure:"))                      Type = input("Enter Type:")                      st = "insert into players values('{0}','{1}','{2}',{3},'{4}','{5}',{6},{7})".format(player\_id,  player\_name,nationality,age,current\_team,speciality,no\_matches,price)                      stbowlinsert= "insert into bowlers values('{0}',{1},{2},{3},{4},'{5}')".format(player\_id,economy  ,wickets\_taken,bowling\_avg,bestfigure,Type)                      cursor.execute(st)                      cursor.execute(stbowlinsert)                      connect.commit()                      break                  elif speciality == 'batsmen':                      avg\_runs = int(input("Enter avg runs:"))                      strikerate = int(input("Enter strike rate:"))                      preffered\_position = input("Enter Preffered position:")                      bestscore = int(input("Enter bestscore:"))                      runs = int(input("Enter Runs:"))                      st = "insert into players values('{0}','{1}','{2}',{3},'{4}','{5}',{6},{7})".format(player\_id,                                                                                                                player\_name,                                                                                                                nationality,                                                                                                                age,                                                                                                                current\_team,                                                                                                                speciality,                                                                                                                no\_matches,price)                      stbatsinsert = "insert into batsmen values({0},{1},'{2}',{3},'{4}',{5})".format(avg\_runs,  strikerate,preffered\_position,bestscore,player\_id,runs)                      cursor.execute(st)                      cursor.execute(stbatsinsert)                      connect.commit()                      break                  elif speciality == "AR":                      avg\_runs = int(input("Enter avg runs:"))                      strikerate = int(input("Enter strike rate:"))                      preffered\_position = (input("Enter Preffered position:"))                      bestscore = int(input("Enter bestscore:"))                      economy = int(input("Enter economy:"))                      wickets\_taken = int(input("Enter wickets taken:"))                      bowling\_avg = int(input("Enter bowling avg:"))                      Type = input("Enter Type:")                      price = int(input("Enter price:"))                      runs = int(input("Enter Runs:"))                      bestfigure = int(input("Enter best figure:"))                      st = "insert into players values('{0}','{1}','{2}',{3},'{4}','{5}',{6},{7})".format(player\_id,                                                                                                                player\_name,                                                                                                                nationality,                                                                                                                age,                                                                                                                current\_team,                                                                                                                speciality,                                                                                                                no\_matches,price)                      stbowlinsert = "insert into bowlers values('{0}',{1},{2},{3},{4},'{5}')".format(player\_id, economy,                                                                                                wickets\_taken,                                                                                                bowling\_avg, bestfigure,Type)                      stbatsinsert = "insert into batsmen values({0},{1},'{2}',{3},'{4}',{5})".format(avg\_runs,  strikerate,                                                                                                    preffered\_position,                                                                                                    bestscore,player\_id,runs)                      cursor.execute(st)                      cursor.execute(stbowlinsert)                      cursor.execute(stbatsinsert)                      connect.commit()                      break                  else:                      print("Incorrect speciality")                      break              break          elif Action == "Modify":              while True:                  import mysql.connector as sql                  #import Player\_search                  connect = sql.connect(host="localhost", user="root", passwd="shriram123",  database="sample")                  cursor = connect.cursor()                  player\_id = input("Enter character id:")                  st = "select speciality from players where playerid = '{0}'".format(player\_id)                  cursor.execute(st)                  k = cursor.fetchone()                  speciality = str(k[0])                  if speciality == "Bowler":                      economy = int(input("Enter economy:"))                      wickets\_taken = int(input("Enter wickets taken:"))                      bowling\_avg = int(input("Enter bowling avg:"))                      bestfigure = int(input("Enter best figure:"))                      Type = input("Enter Type:")                      stbowlmodify= '''update bowlers                      set economy = {0},wickets\_taken = {1},bowling\_avg = {2},bestfigure = {3},type\_ = '{4}'                      where playerid = "{5}"'''.format(economy,wickets\_taken,bowling\_avg,bestfigure,  Type,player\_id)                      cursor.execute(stbowlmodify)                      connect.commit()                      break                  elif speciality == 'batsmen':                      avg\_runs = int(input("Enter avg runs:"))                      strikerate = int(input("Enter strike rate:"))                      preffered\_position = input("Enter Preffered position:")                      bestscore = int(input("Enter bestscore:"))                      runs = int(input("Enter Runs:"))                      stbatmodify= '''update batsmen                       set avg\_runs = {0},strikerate = {1},preffered\_position = '{2}',bestscore = {3},runs ={4}                       where (playerid = '{5}')'''.format(avg\_runs,strikerate,preffered\_position,bestscore,runs,  player\_id)                      cursor.execute(stbatmodify)                      connect.commit()                      break                  elif speciality == "AR":                      avg\_runs = int(input("Enter avg runs:"))                      strikerate = int(input("Enter strike rate:"))                      preffered\_position = (input("Enter Preffered position:"))                      bestscore = int(input("Enter bestscore:"))                      Runs - int(input("Enter the Runs:"))                      economy = int(input("Enter economy:"))                      wickets\_taken = int(input("Enter wickets taken:"))                      bowling\_avg = int(input("Enter bowling avg:"))                      bestfigure = int(input("Enter best figure:"))                      Type = input("Enter Bowling type 'Pace' or'spin':")                      stbatmodify='''update batsmen                       set avg\_runs = {0},strikerate = {1},preffered\_position = {2},runs = {5},bestscore = {3}                       where (playerid = '{4}')'''.format(avg\_runs,strikerate,preffered\_position,bestscore,player\_id,  Runs)                      stbowlmodify = '''update bowlers                      set economy = {0},wickets\_taken = {1},bowling\_avg = {2},bestfigure = {3},type\_ = '{5}'                      where playerid = "{4}"'''.format(economy,wickets\_taken,bowling\_avg,bestfigure,player\_id,  Type)                      cursor.execute(stbowlmodify)                      cursor.execute(stbatmodify)                      connect.commit()                      break              break          elif Action == "Delete":              while True:                  import mysql.connector as sql                  import Player\_search                  connect = sql.connect(host="localhost", user="root", passwd="shriram123",  database="sample")                  cursor = connect.cursor()                  player\_id = input("Enter character id:")                  st = "delete from players where playerid = '{0}'".format(player\_id)                  stbowldelete = "delete from bowlers where playerid = '{0}'".format(player\_id)                  stbatdelete = "delete from batsmen where playerid = '{0}'".format(player\_id)                  cursor.execute(st)                  cursor.execute(stbowldelete)                  cursor.execute(stbatdelete)                  connect.commit()                  break          else:              print("Please enter a given choice your choice was invalid")  def player\_search():      import mysql.connector as sql      import Player\_search      connect = sql.connect(host="localhost", user="root", passwd="shriram123", database="sample")      cursor = connect.cursor()      stbat = ""      stbowl = ""      print("Enter the necessary stats:")      age = int(input("Enter required age:"))      while True:          speciality = input("Enter 'AR' or 'Bowler' or batsmen:")          if speciality not in ("AR", "Bowler", "batsmen"):              print("Incorrect speciality enter again")              continue          else:              break      if speciality in ("batsmen", "AR"):          avg\_runs = int(input("Enter required avg runs per match:"))          strikerate = int(input("Enter required strike rate:"))          preffered\_position = int(input("Enter Preffered position"))          stbat = '''select \* from players,batsmen                          where (avg\_runs between {0} and {1}) and                         (strikerate between {2} and {3}) and                         (preffered\_position between {4} and {5}) and                         players.playerid = batsmen.playerid '''.format(avg\_runs - 100, avg\_runs + 100,  strikerate - 100,                                                                     strikerate + 1000, preffered\_position - 50,  preffered\_position + 500)      if speciality in ("Bowler", "AR"):          economy = int(input("Enter required economy:"))          bowling\_avg = int(input("Enter required bowling average:"))          Type = input("Enter required Type:")          stbowl = '''select \* from players,bowlers                        where (economy between {0} and {1}) and                        (bowling\_avg between {2} and {3}) and                        (type\_ = "{4}") and                        bowlers.playerid = players.playerid'''.format(economy - 100, economy + 100,  bowling\_avg - 500, bowling\_avg + 500,                                                                  Type)      if stbowl != "" and stbat != "":          stboth = '''select \* from players,batsmen,bowlers                             where (avg\_runs between {0} and {1}) and                         (strikerate between {2} and {3}) and                         (preffered\_position between {4} and {5}) and                          (economy between {6} and {7}) and                        (bowling\_avg between {8} and {9}) and                        (type\_ = "{10}") and                         players.playerid = batsmen.playerid and batsmen.playerid = bowlers.playerid                         '''.format(avg\_runs - 100, avg\_runs + 100, strikerate - 100,                                                                     strikerate + 1000, preffered\_position - 50,  preffered\_position + 500,economy - 100,                                                                     economy + 100, bowling\_avg - 500, bowling\_avg + 500,Type)          print('playername', 'nationality', 'age', 'current\_team', 'speciality', 'no\_matches','Price'                , sep='|', end="|")          print('avg\_runs', 'strikerate', 'preffered\_position', 'bestcore','Runs','', sep='|',end="")          print("economy", "wicket taken", "bowling avg", 'bestfigure','Type', sep='|')          cursor.execute(stboth)          n = 1          for i in cursor:              print(n,")",sep="",end="")              n = n + 1              for j in i[1:12]:                  print(j,end = "|")              for k in i[13:14]:                  print(k, end="|")              for l in i[15:20]:                  print(l, end="|")              print()      elif stbat != "" and stbowl == "":          print('playername', 'nationality', 'age', 'current\_team', 'speciality','no\_matches',"price"          ,sep = '|', end = "")          print('avg\_runs', 'strikerate', 'preffered\_position', 'bestcore','runs', sep='|')          cursor.execute(stbat)          n = 1          for i in cursor:              print(n, ")", sep="", end="")              n = n + 1              for j in i[1:12]:                  print(j,end = "|")              for k in i[13:14]:                  print(k,end="|")              print("\b")      else:          print('playername', 'nationality', 'age', 'current\_team', 'speciality', 'no\_matches', "price"                , sep='|', end="")          print('economy', "wicket taken", "bowling avg","bestfigure",sep = "|")          cursor.execute(stbowl)          n = 1          for i in cursor:              print(n, ")", sep="", end="")              n = n + 1              for j in i[1:12]:                  print(j, end="|")              for k in i[13:14]:                  print(k, end="|")              print("\b")  def password(Type):      import mysql.connector as sql      connect = sql.connect(host="localhost", user="root", passwd="shriram123", database="sample")      cursor = connect.cursor()      while True:          username = input("Enter your username:")          cursor.execute("Select \* from passwrd")          password = input("Enter your password:")          for i in cursor:              if i[1] == username and i[2] == password and i[3] == Type:                  print("Hello", username)                  return username                  break          else:              print("Incorrect login credentials,TRY AGAIN")              continue          break   |  | | --- | |  | |  |

**RELATED TABLES AND OUTPUT**

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Graphical user interface, text

Description automatically generated

Text

Description automatically generated

A screenshot of a computer

Description automatically generated

Graphical user interface, text

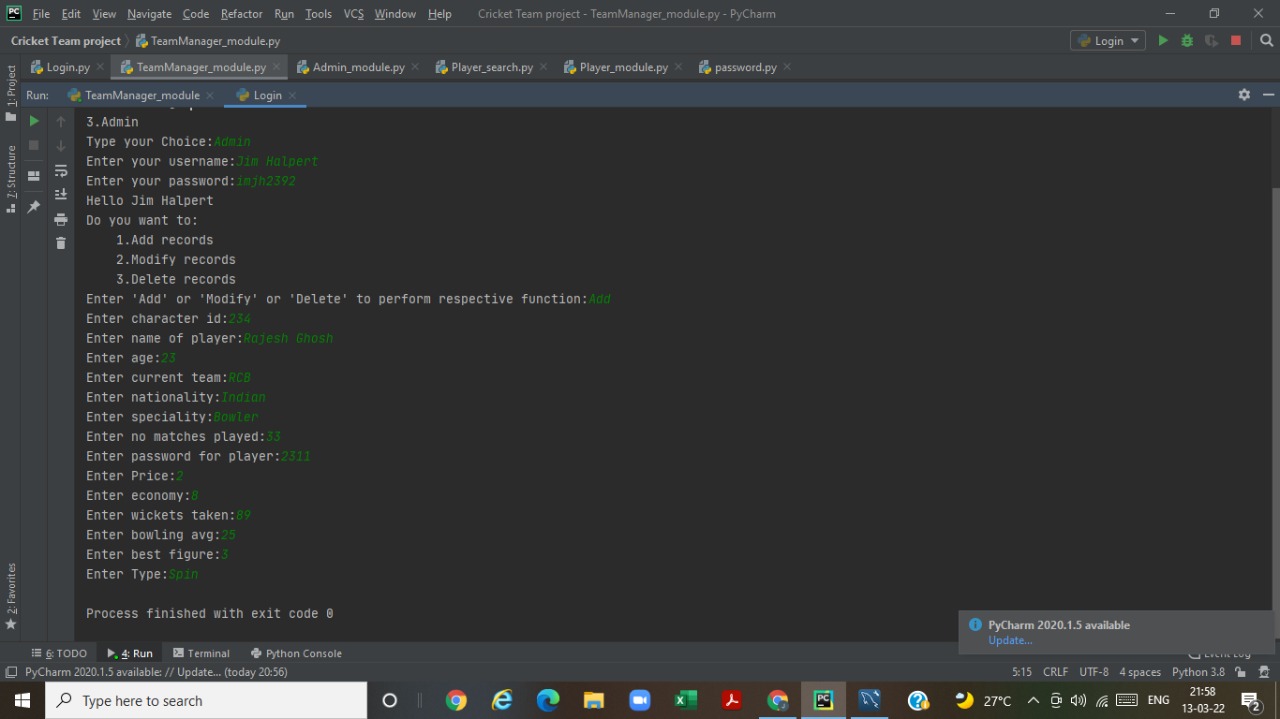
Description automatically generated

A screenshot of a computer

Description automatically generated

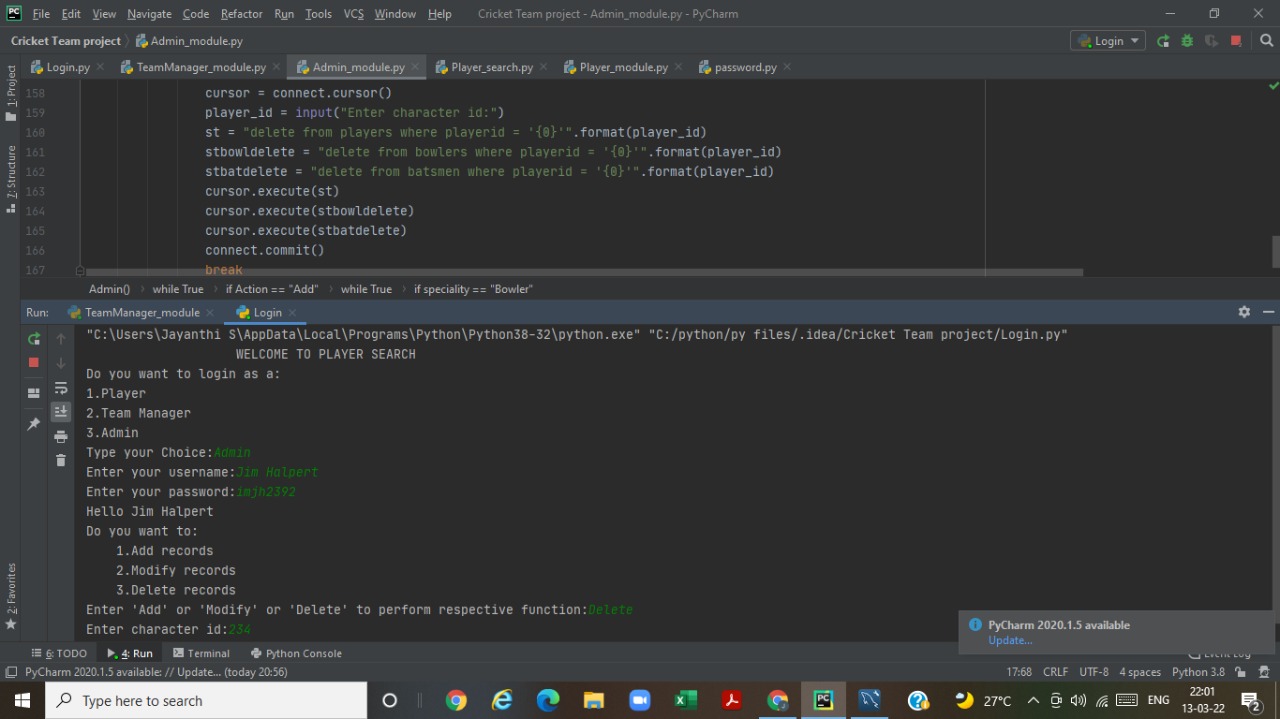
Text

Description automatically generated



Text

Description automatically generated



**PROJECT IMPACT**

* The proposed project would be completely automated
* Less effort is required for maintaining the database of players using this software
* Margin of error will be reduced and choosing members will be a breeze by using this software
* The end user, be it an IPL team owner or a fantasy game user, or both can skim through a guide of options and choose the best player
* The managerial functions are carried out with ease by the admin. Admin can add/delete or modify cricket player details

**FUTURE SCOPE AND DEVELOPMENT**

* This work can be extended in the future in such a way that the database management system can make use of machine learning-based algorithms that predicts the cost at which a player can be sold in the auctions
* The players' selling price can be estimated using their past performance parameters like runs, balls, innings, wickets and matches played via machine learning regression models
* The exponential rise in popularity of fantasy games in India also has paved way for a new market that could benefit from such digital infrastructures.

**BIBLIOGRAPHY**

* [www.python.org](http://www.python.org)
* [www.geeksforgeeks.com](http://www.geeksforgeeks.com)
* [www.programiz.com](http://www.programiz.com)
* The Complete Reference Python by Martin C. Brown from Mc. Grawhill publication
* Computer Science with Python by Sunita Arora from Dhan Patrai & co. publication