**BASAVARAJESWARI GROUP OF INSTITUTIONS**



# BALLARI INSTITUTE OF TECHNOLOGY & MANAGEMENT

**AUTONOMOUS INSTITUTE UNDER VISVESVARAYA TECHNOLOGICAL UNIVERSITY**

**JNANA SANGAMA, BELAGAVI 590018**

**DEPARTMENT OF ELECTRICAL & ELECTRONICS ENGINEERING**

**Seminar Report on**

# “Sports league management system”

# For the course: Python

**Name Usn no**

**Bhoomika 3BR23EC025**

**Bhumika.c.k 3BR23EC027**

**Manasa.M 3BR23EC091**

**Waseema kousar 3BR23EC184**

BALLARI INSTITUTE OF TECHNOLOGY & MANAGEMENT

NACC Accredited institution\*

(Recognised by Govt.of karnataka,approved by AICTE,NewDelhi & Affiliated to Visvesvaraya

Technological University, Belgavi)

“Jnana Gangotri” Campus,No.873/2,Ballari-Hospet Road,Allipur,Ballari-583104

Karnataka,India.

Ph: 08392-237100/23719, Fax:08392-237197

2024-2025

**S**

**ports league management system**

****

**Introduction :**

*Managing a sports league involves handling numerous tasks such as team registrations, scheduling matches, tracking player statistics, and ensuring smooth communication among participants. A robust and efficient sports league management system can significantly ease these operations. This document introduces a Python-based sports league management system designed to streamline the administration of sports leagues, providing a comprehensive solution for organizers, teams, and players.*

**Problem statement:**

*League Management System that allows users to manage a list of teams in a sports league. Sports The system should enable users to perform basic operations such as adding, viewing, updating, and deleting teams. Each team will have the following attributes:*

***Domain:***

*A system designed to manage schedules , teams , scores , and logistics for organized sports leagues efficiently*

**Objectives :**

*The objective of a sports league management system is to efficiently manage teams, players, and match schedules. It streamlines tasks like adding, updating, and deleting information, recording scores, and generating performance reports, enhancing transparency and decision-making for league administrators, coaches, and players.*

**Algorithm:**

1. Initialize league as an empty list.

*2. Define add\_team function : Create an empty dictionary team . Prompt user for team details and store in team . Append team to league . Print success message.*

*3. Define display\_teams function : If league is empty, print message and return .Otherwise, print details of each team in league.*

*4. Define update\_team function : Call display\_teams . Prompt user for team number to update . If valid, prompt user for new details and update selected team . Print success message.*

*5. Define delete\_team function : Call display\_teams . Prompt user for team number to delete . If valid, remove selected team from league . Print success message.*

*6. Define menu function : Enter infinite loop : Print menu options . Prompt user for choice . Call corresponding function or exit based on choice.*

*7. If script is run directly, call menu function*

**Module split up:**

* ***Create():****the program allows user to create new teams and record their mistakes*
* ***Update():****the program automatically update the teams records when matches are added*
* ***Display():****he program displays the current records of all items*

**Conclustion**:

*The Python-based sports league management system enhances league administration by automating scheduling, statistics tracking, and communication. Utilizing frameworks like Django or Flask and databases such as SQLite or PostgreSQL, it offers scalability and reliability. This user-friendly system improves efficiency and transparency, allowing organizers to focus on fostering competitive sports environments, ultimately revolutionizing sports management with technology*

**Future Enhancements:**

*Enhance the sports league management system by adding mobile app integration, advanced analytics, secure payment processing, live streaming, customization options, and social media integration. These features will increase accessibility, functionality, and user engagement, making the system more versatile and user-friendly.*

class Team:

    def \_\_init\_\_(self, name, captain, players,score):

        self.name = name

        self.captain = captain

        self.players = players

        self.score=score

    def update(self, name=None, captain=None, players=None,score=None):

        if name:

            self.name = name

        if captain:

            self.captain = captain

        if players:

            self.players = players

        if score:

            self.score=score

class League:

    def \_\_init\_\_(self):

        self.teams = []

    def add\_team(self):

        name = input("Enter team name: ")

        captain = input("Enter the captain name: ")

        score=input("Enter the score:")

        while True:

            try:

                players = int(input("Enter number of players: "))

                break

            except ValueError:

                print("Please enter a valid number for players:")

        new\_team = Team(name, captain, players,score)

        self.teams.append(new\_team)

        print(f"Team {name} added successfully!\n")

    def display\_teams(self):

        if not self.teams:

            print("No teams in the league yet.")

            return

        print("\nTeams in the League:")

        for i, team in enumerate(self.teams):

            print(f"Team {i + 1}: {team.name}, Captain: {team.captain}, Players: {team.players},score:{team.score}")

    def update\_team(self):

        self.display\_teams()

        try:

            team\_index = int(input("\nEnter the number of the team you want to update: ")) - 1

            if 0 <= team\_index < len(self.teams):

                team = self.teams[team\_index]

                name = input(f"Enter new name for {team.name}:")

                captain = input(f"Enter new captain for {name}:")

                score=input(f"Enter the new score for{name}:")

                while True:

                    players = input(f"Enter new number of players for {name} : ")

                    if players:

                        try:

                            players = int(players)

                            break

                        except ValueError:

                            print("Please enter a valid number for players.")

                    else:

                        players = team.players

                        break

                team.update(name,captain, players,score)

                print(f"Team {team.name} updated successfully!\n")

            else:

                print("Invalid team number.")

        except ValueError:

            print("Invalid input. Please enter a valid number.")

    def delete\_team(self):

        self.display\_teams()

        try:

            team\_index = int(input("\nEnter the number of the team you want to delete: ")) - 1

            if 0 <= team\_index < len(self.teams):

                team = self.teams.pop(team\_index)

                print(f"Team {team.name} deleted successfully!\n")

            else:

                print("Invalid team number.")

        except ValueError:

            print("Invalid input. Please enter a valid number.")

def menu():

    league = League()

    while True:

        print("\nSports League Management System")

        print("1. Add Team")

        print("2. Display Teams")

        print("3. Update Team")

        print("4. Delete Team")

        print("5. Exit")

        choice = input("Enter your choice (1-5): ")

        if choice == '1':

            league.add\_team()

        elif choice == '2':

            league.display\_teams()

        elif choice == '3':

            league.update\_team()

        elif choice == '4':

            league.delete\_team()

        elif choice == '5':

            print("Exiting the system.")

            break

        else:

            print("Invalid choice. Please try again.")

if \_\_name\_\_ == "\_\_main\_\_":

    menu()

**Result:**

Sports League Management System

1. Add Team

2. Display Teams

3. Update Team

4. Delete Team

5. Exit

Enter your choice (1-5): 1

Enter team name: RCB

Enter the captain name: Virat Kohli

Enter the score:278

Enter number of players: 11

Team RCB added successfully!

Sports League Management System

1. Add Team

2. Display Teams

3. Update Team

4. Delete Team

5. Exit

Enter your choice (1-5): 1

Enter team name: SRH

Enter the captain name: Pat Cummin

Enter the score:150

Enter number of players: 11

Team SRH added successfully!

Sports League Management System

1. Add Team

2. Display Teams

3. Update Team

4. Delete Team

5. Exit

Enter your choice (1-5): 1

Enter team name: LSG

Enter the captain name: KL Rahul

Enter the score:200

Enter number of players: 11

Team LSG added successfully!

Sports League Management System

1. Add Team

2. Display Teams

3. Update Team

4. Delete Team

5. Exit

Enter your choice (1-5): 2

Teams in the League:

Team 1: RCB, Captain: Virat Kohli, Players: 11,score:278

Team 2: SRH, Captain: Pat Cummin, Players: 11,score:150

Team 3: LSG, Captain: KL Rahul, Players: 11,score:200

Sports League Management System

1. Add Team

2. Display Teams

3. Update Team

4. Delete Team

5. Exit

Enter your choice (1-5): 3

Teams in the League:

Team 1: RCB, Captain: Virat Kohli, Players: 11,score:278

Team 2: SRH, Captain: Pat Cummin, Players: 11,score:150

Team 3: LSG, Captain: KL Rahul, Players: 11,score:200

Enter the number of the team you want to update: 2

Enter new name for SRH:CSK

Enter new captain for CSK:MS Dhoni

Enter the new score forCSK:220

Enter new number of players for CSK : 11

Team CSK updated successfully!

Sports League Management System

1. Add Team

2. Display Teams

3. Update Team

4. Delete Team

5. Exit

Enter your choice (1-5): 2

Teams in the League:

Team 1: RCB, Captain: Virat Kohli, Players: 11,score:278

Team 2: CSK, Captain: MS Dhoni, Players: 11,score:220

Team 3: LSG, Captain: KL Rahul, Players: 11,score:200

Sports League Management System

1. Add Team

2. Display Teams

3. Update Team

4. Delete Team

5. Exit

Enter your choice (1-5): 4

Teams in the League:

Team 1: RCB, Captain: Virat Kohli, Players: 11,score:278

Team 2: CSK, Captain: MS Dhoni, Players: 11,score:220

Team 3: LSG, Captain: KL Rahul, Players: 11,score:200

Enter the number of the team you want to delete: 3

Team LSG deleted successfully!

Sports League Management System

1. Add Team

2. Display Teams

3. Update Team

4. Delete Team

5. Exit

Enter your choice (1-5): 2

Teams in the League:

Team 1: RCB, Captain: Virat Kohli, Players: 11,score:278

Team 2: CSK, Captain: MS Dhoni, Players: 11,score:220

Sports League Management System

1. Add Team

2. Display Teams

3. Update Team

4. Delete Team

5. Exit

Enter your choice (1-5): 5

Exiting the system.