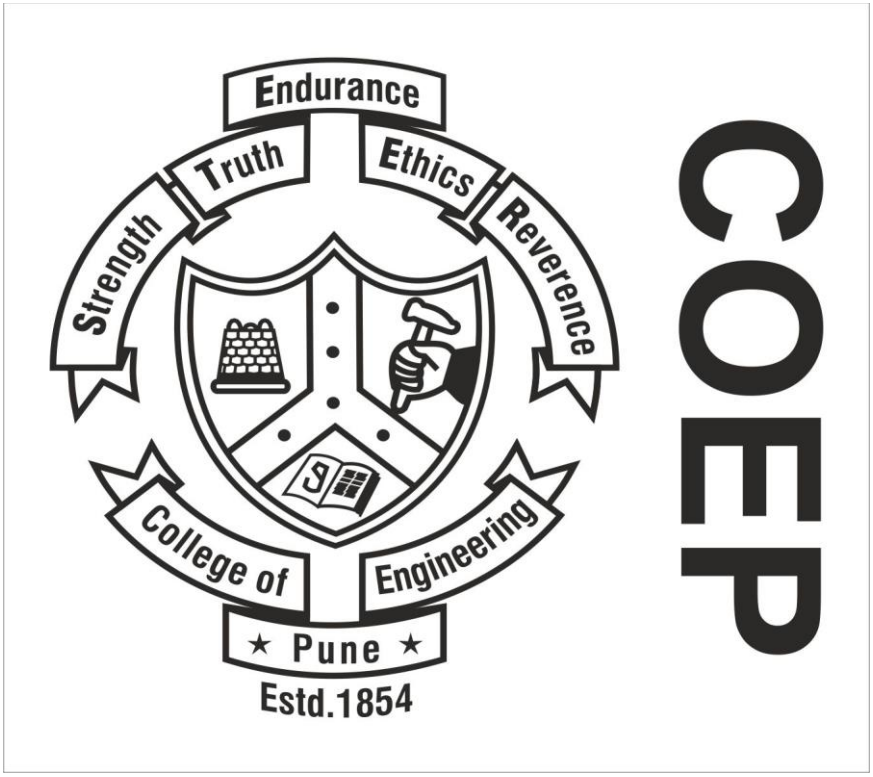
Cricket Statistics Management
Project Report
Software Engineering Mini Project II

Nikhil Repale
MIS - 111903050
repalena19.comp@coep.ac.in

Ritik Wahane
MIS - 111903070
wahanerr19.comp@coep.ac.in

TY Computer Engineering
Department of Computer Engineering, COEP



1. Problem Statement:

Cricket is the most watched sport in India. The craze of cricket in India is huge and people are quite enthusiastic about it. Cricket enthusiasts also like to check player statistics to compare their statistics with other players. If they want to look for statistics, they have to visit different websites and after visiting those many sites, it may happen that they didn't find enough information. It is challenging to find single platform where we can find statistics of all players/team.

2. Objectives:

1. To develop an efficient and user-friendly platform for analysis of cricket statistics.

2. To have an efficient database that can store, retrieve, update and delete statistics built using MySQL.

3. Motivation:

With an aim to solve of problem every cricket enthusiast to find single platform to get all information of player, we will provide user-friendly platform where user can find/compare statistics of all players/team. It will provide user to store, retrieve, update, delete and display the statistics.

4. Summary of SRS:

Cricket Statistics Management should provide web-based UI where users can add or retrieve the existing data from the system. It should give functionality to Insert, Search, Update and Delete. Users can also update the information in database but they cannot increase the fields to be added or make any changes in tables.

4.1 Software Constraints:

a. Frontend: React, CSS
   Backend: Nodejs, MySQL database

4.2 Different options/interface available to user and prerequisite steps before each functionality

1. Home Button:

   - User will have option to add player, add team, add league, add match and search.

2. Player Button:

   - User can add player here.

3. Team Button:

   - User can add team here.

4. Search Button:

   - User can search teams and players if they are present in database
   - Also, they can view details of players and teams by clicking on "Details"

5. Add League Button:

   - User can add leagues here. Already existing league will be visible there.
   - User can update or delete league

6. Add Match Button:

   - To insert matches we will need 2 or more teams and at-least 1 league

7. Add Season Button:

- To insert season, team and league should be already present in database
- After clicking “Details” button of league, user can add season
- User can update and delete season

8. Add Ranks:

- We can add players after adding season by clicking “Details” button followed by “Add Ranks” button. Teams should be already added to database if we want to insert rank.

9. Player_stats:

- After clicking on “Details” button of any team from team rankings user can add player and his statistics. Player should be already present in database to add their statistics. Player whose statistics added can be seen under the team whose ranking user has selected earlier.

4.3 External Interface Requirements

4.3.1 User Interfaces

The software provides good graphical interface for the user so they can operate on the system, performing the required task such as create, viewing the details of the player.

Home Page:

i. User will have option to add League, add season and add match.
ii. User can add leagues like IPL, BBL, PSL, etc.

Player Page:

User can add player, view details of player

Teams Page:

It will give an option for adding name of team in league.

User can view details of team rankings.

Search Page:

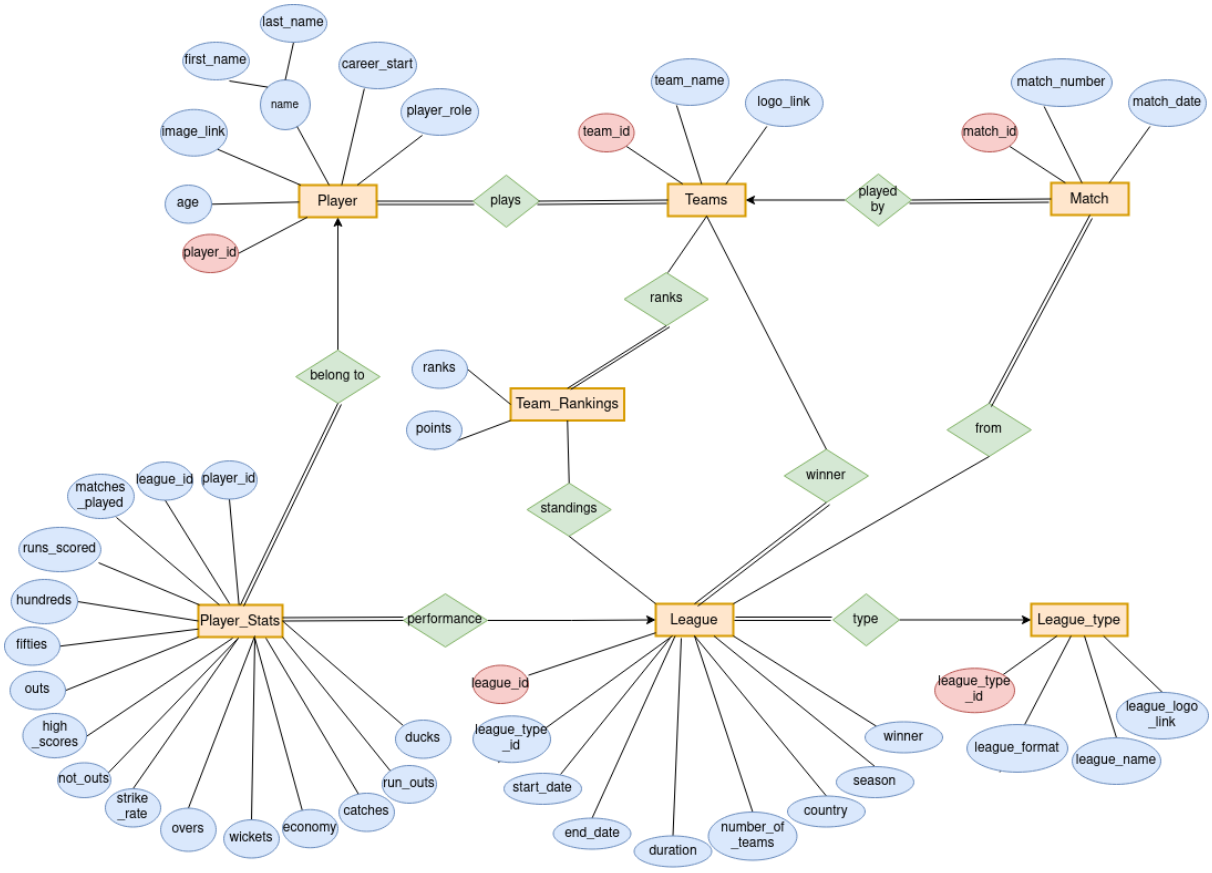
User can search teams and players

Also, they can view details of players and teams

4.3.2 Communication Interfaces

- Web browser to run app on localhost
- To configure NodeJs and MySQL in backend using username and password

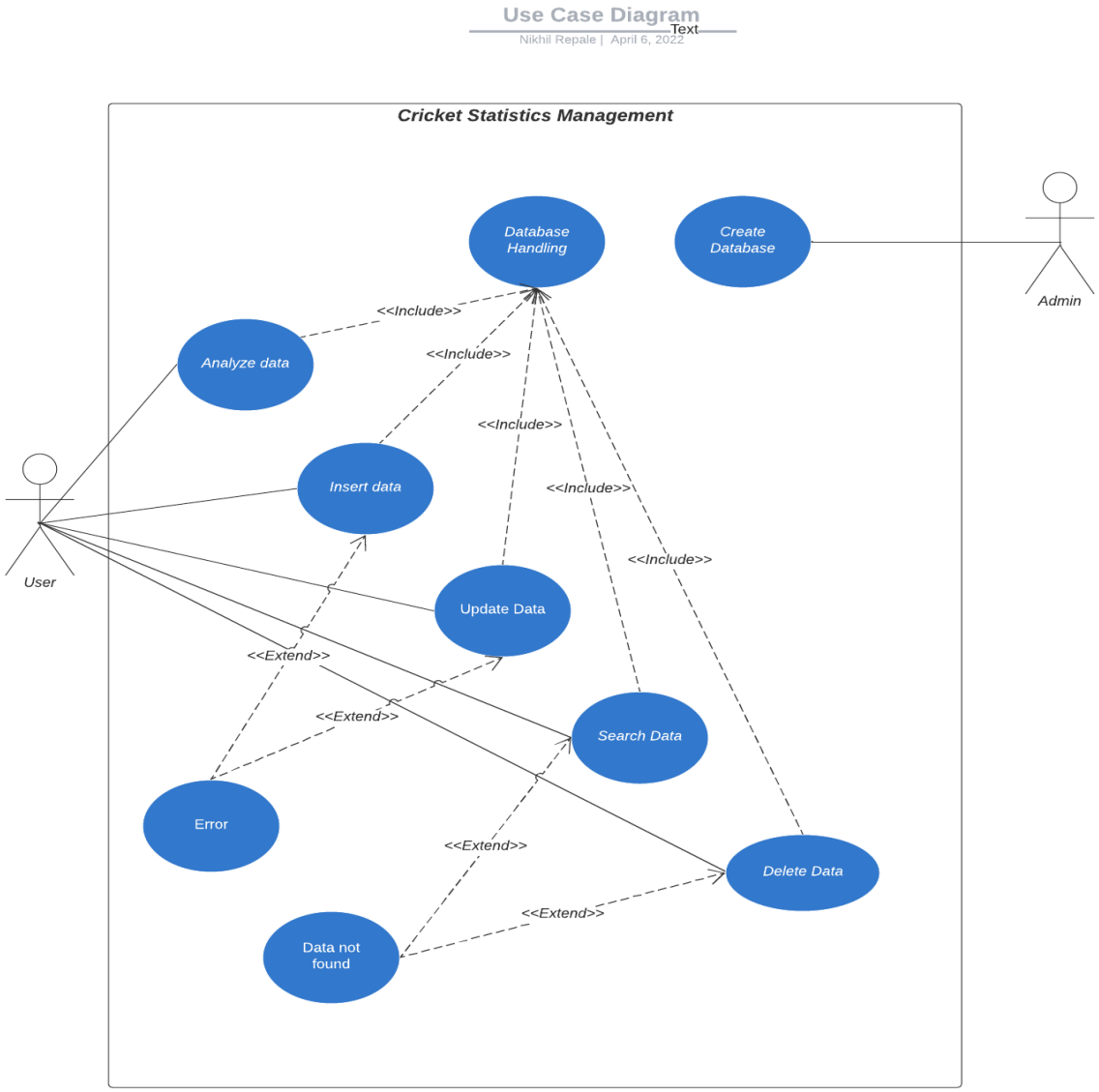
5. ERD:



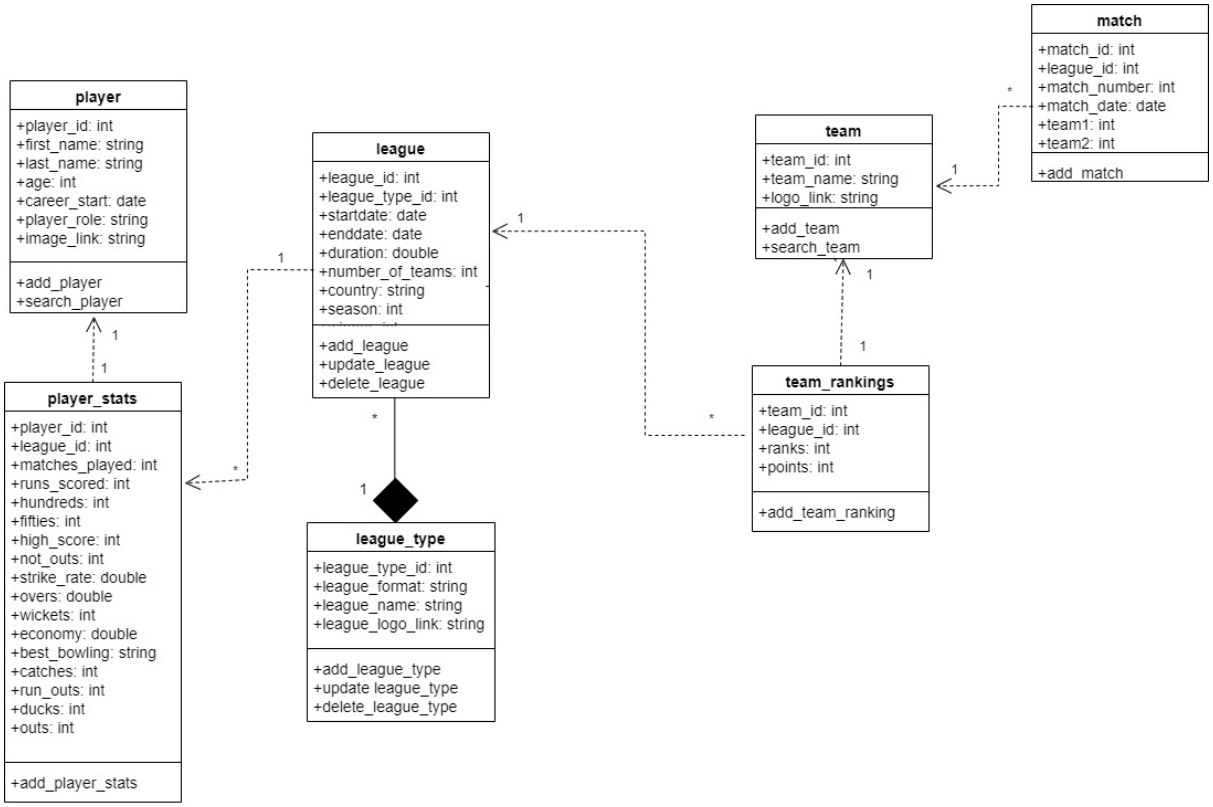
6. UML and explanation:

1. Use case diagram

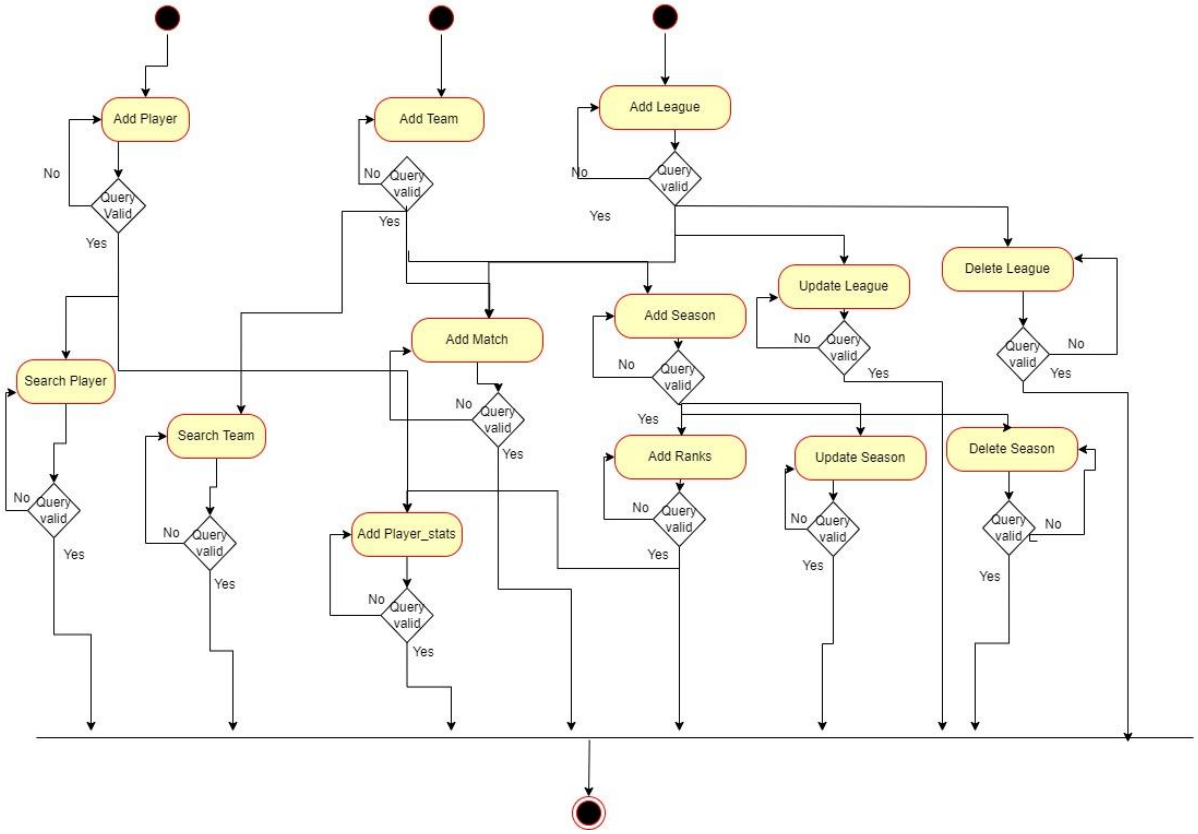
In this diagram our user is an actor. There are five use cases included in database handling viz. Analyze data, Insert data, Update data, Search Data, Delete data.



2. Class diagram



3. Activity Diagram

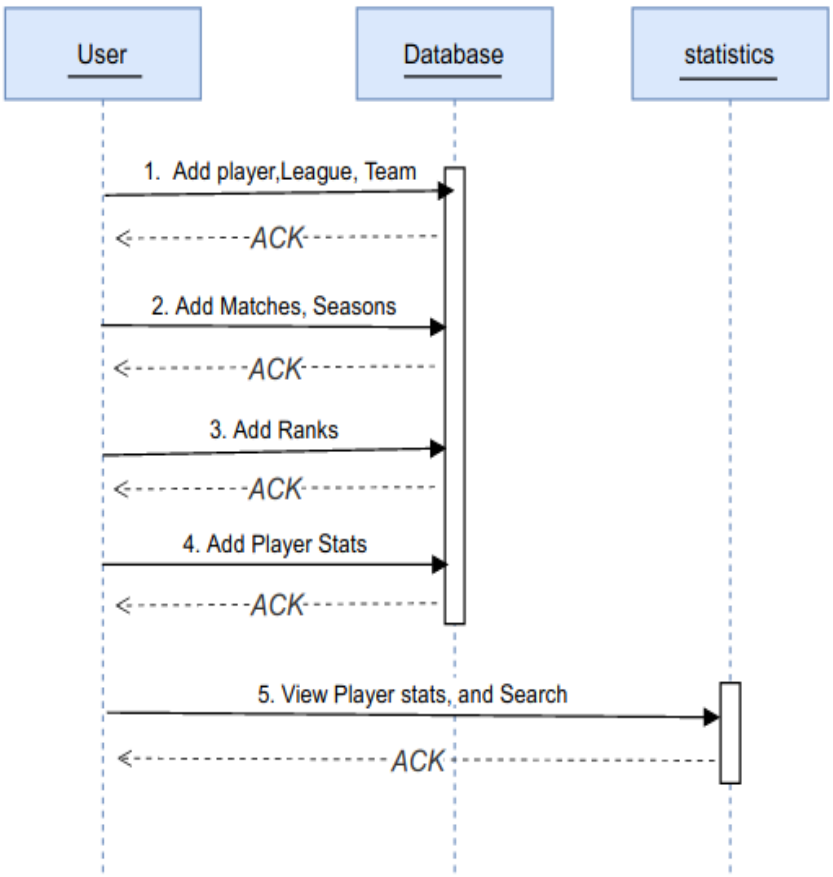


4. State diagram

We will start with adding a player in the database Further we will add team and league data. After having this information we can add match data. Further we will add season and ranks. Last stage is addition of player statistics.

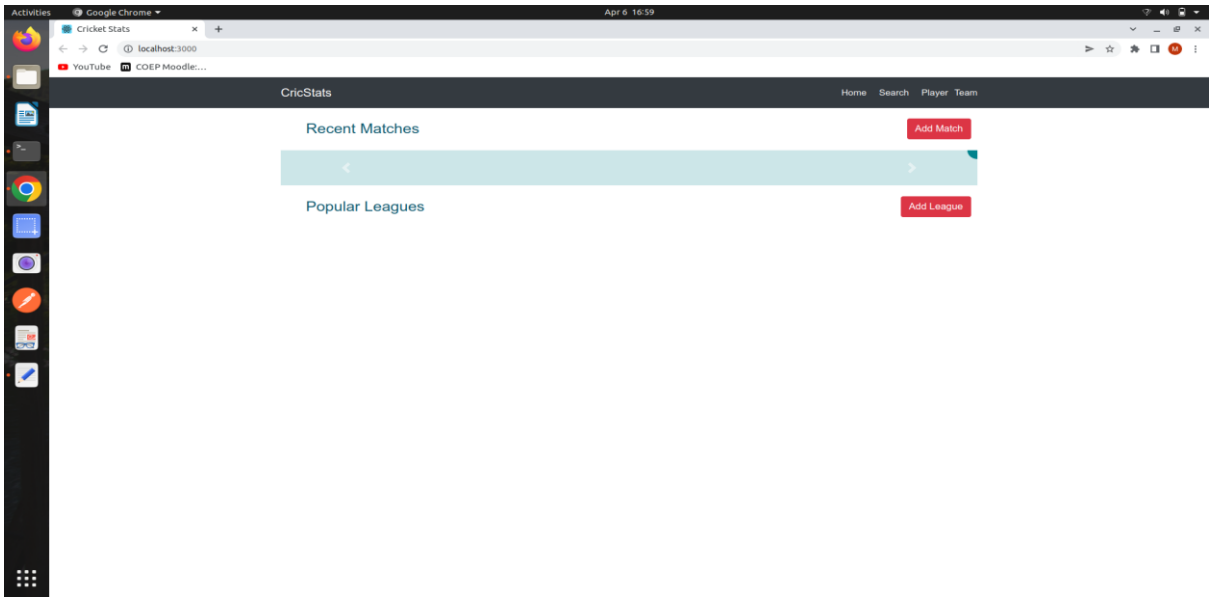
5. Sequence Diagram

User must add player league and team into the database to go further states.
Afterwards he can add matches and seasons data.
Then he is allowed to add rankings of teams.
At last he can add player stats.
After adding data correctly he will receive an acknowledgment from system .
Now he can search and view statistics.

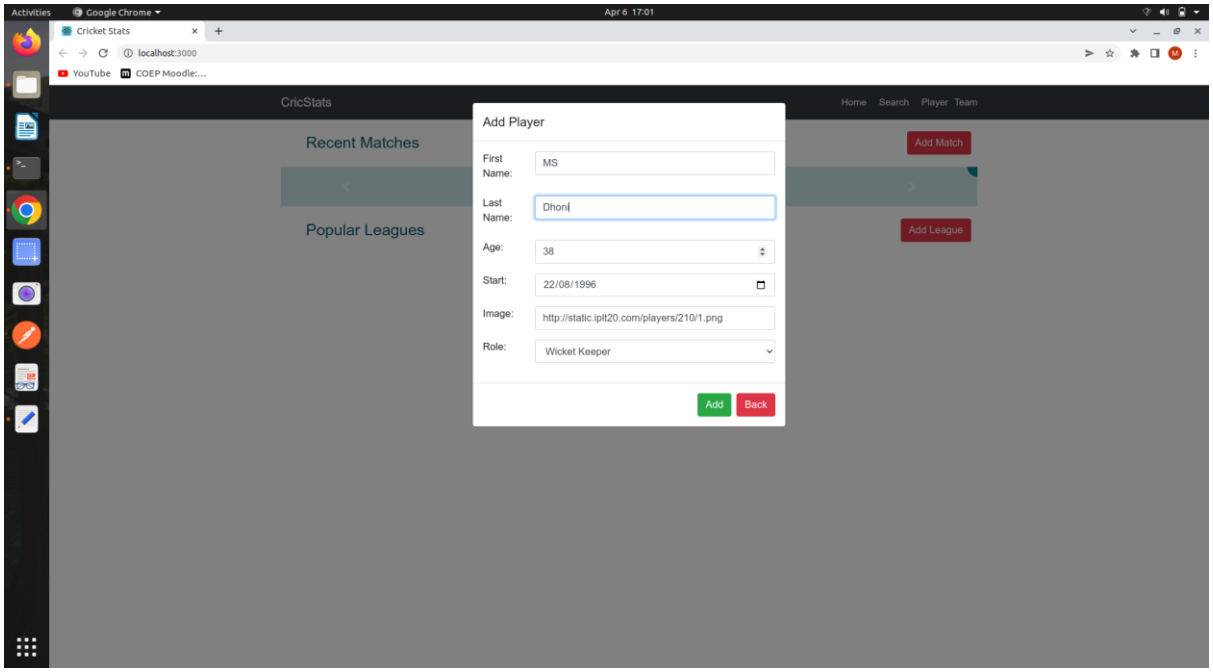


7. Coding Screenshots with Result:

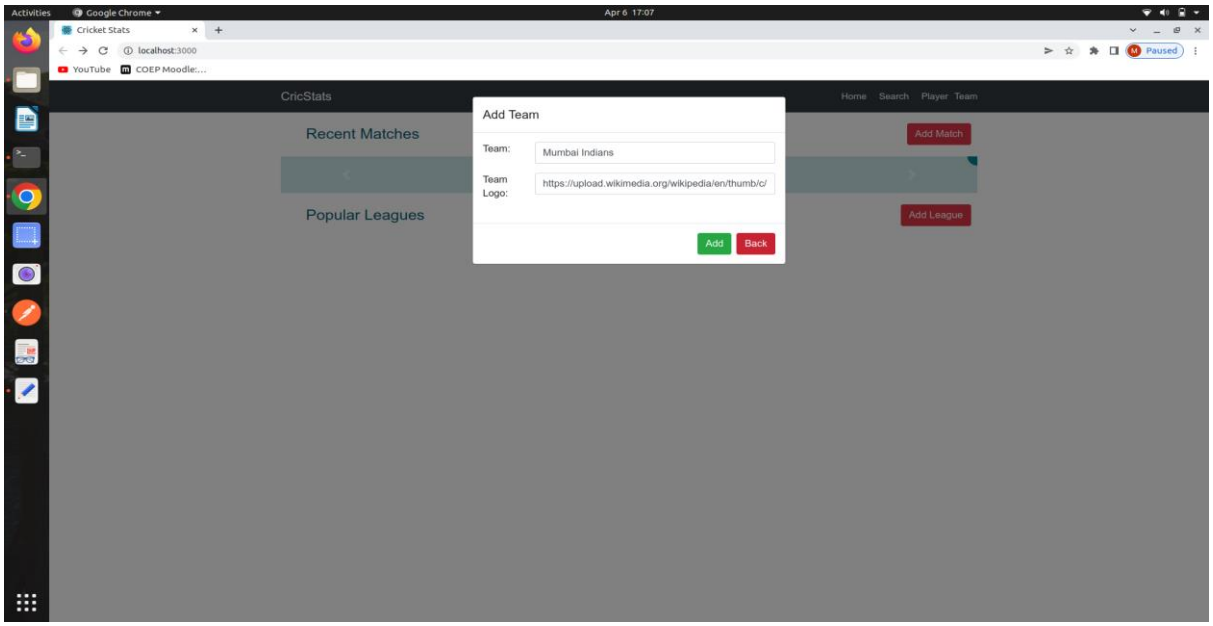
1. Home page



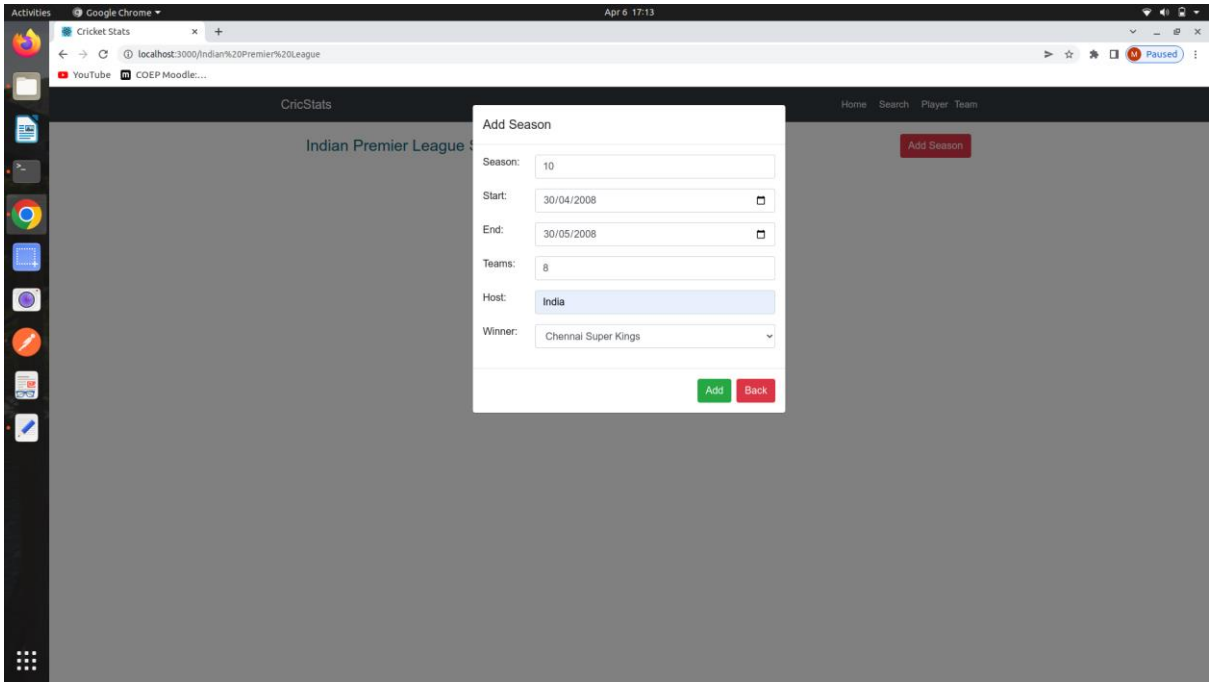
2. Add player



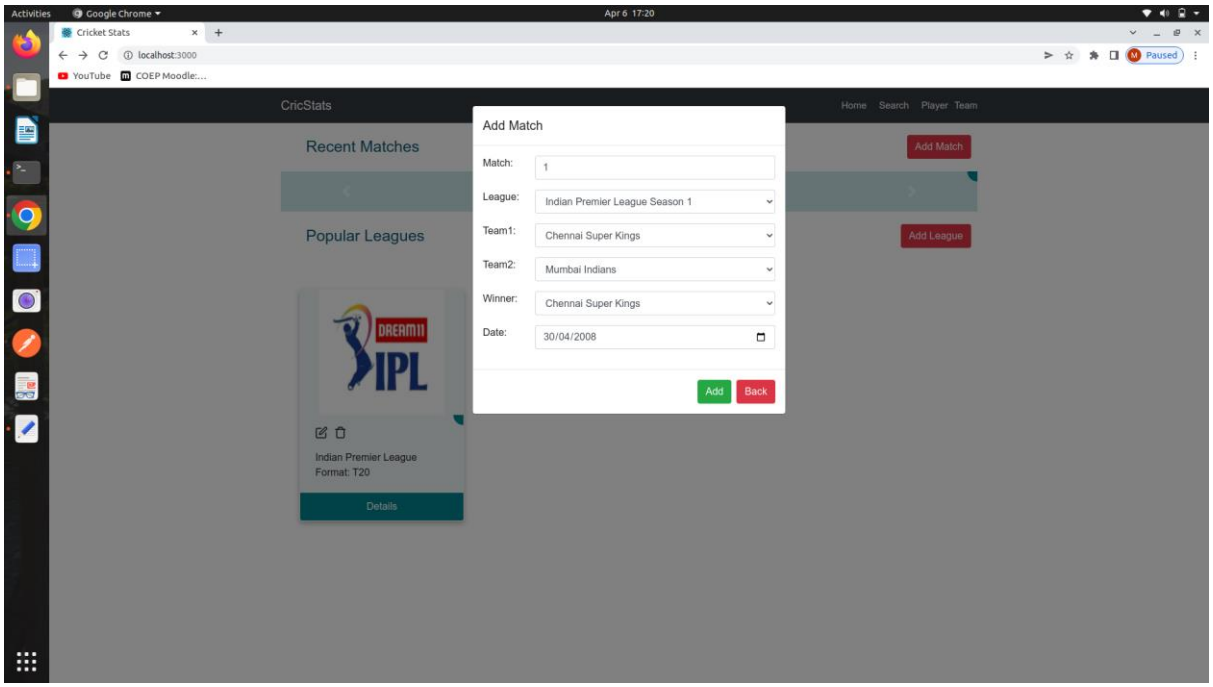
3. Add teams



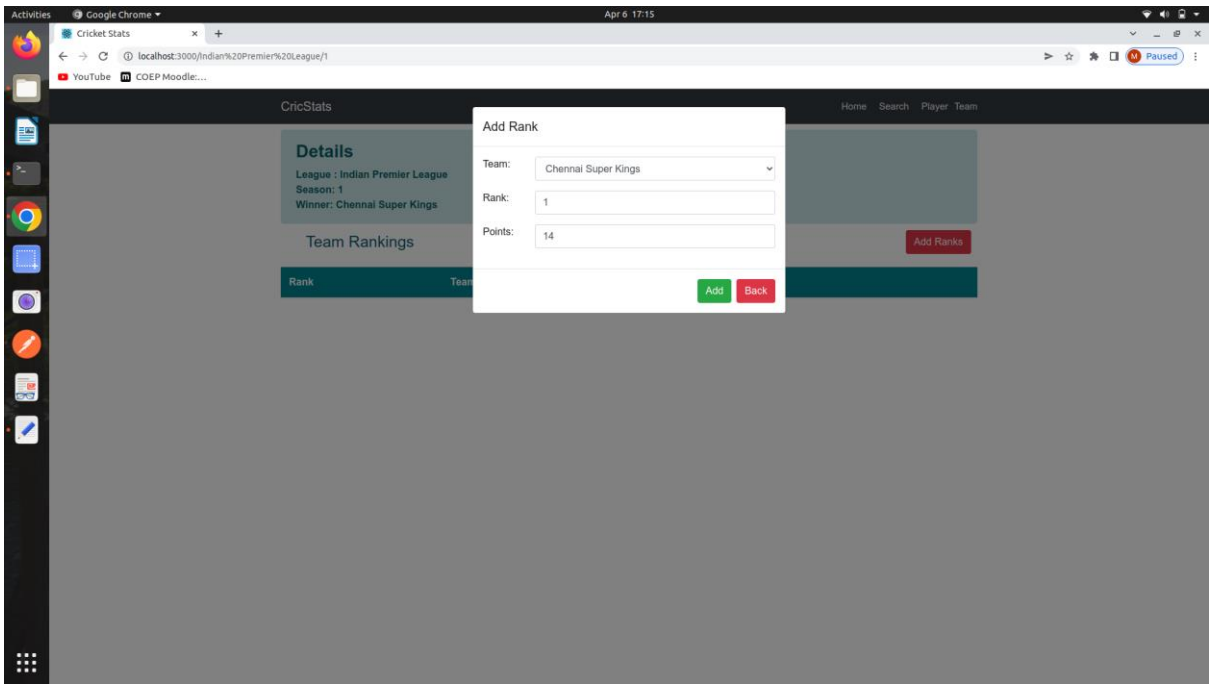
4. Add seasons



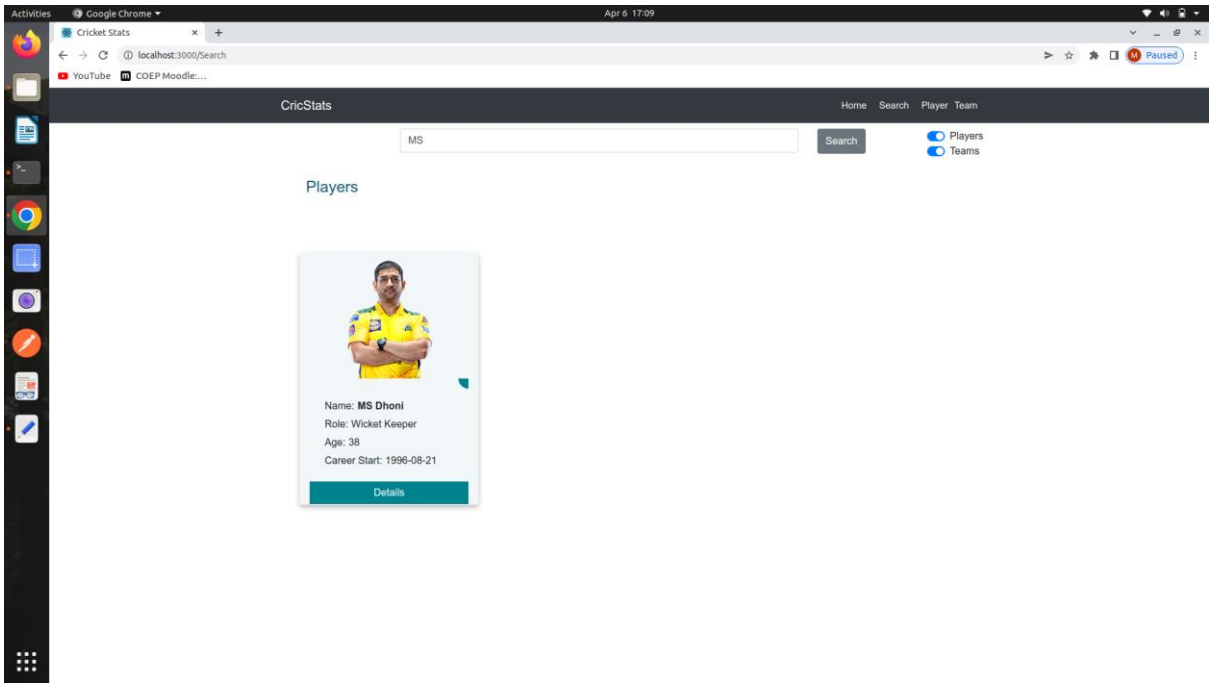
5. Add Match



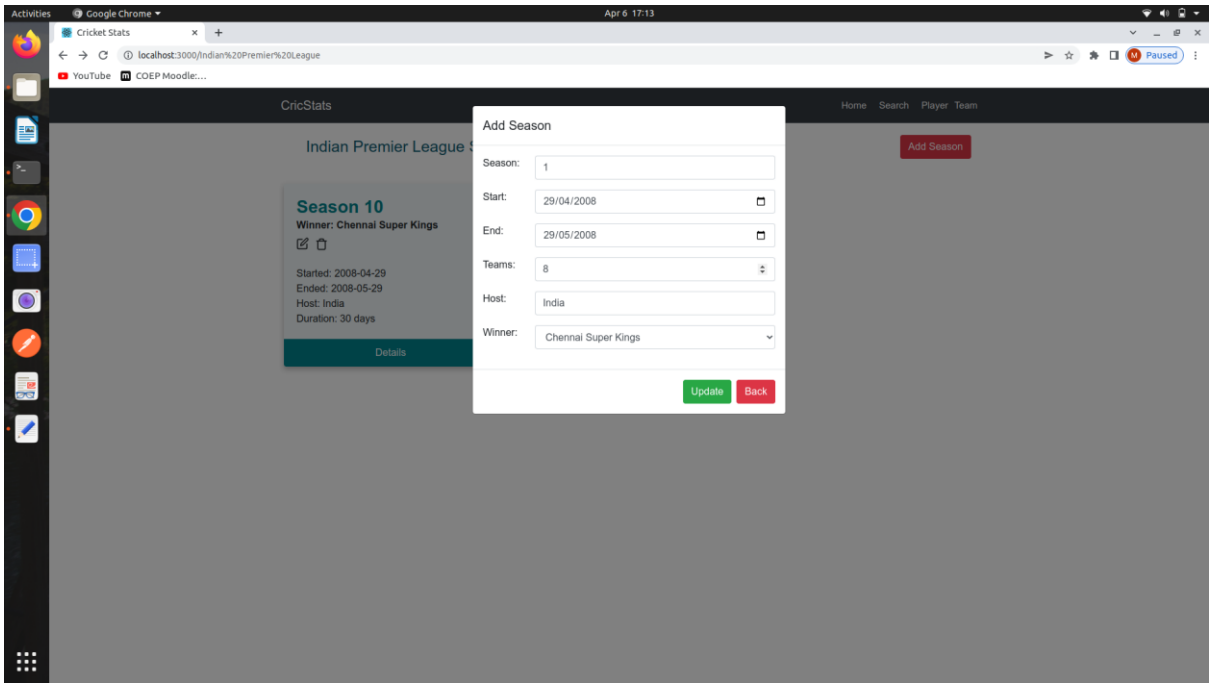
6. Add rankings



7. Search



8. Update Season



8. Testing (approach and bugs found):

Decision table testing

It is used to test system behavior for various inputs.

We map the different input combinations and their corresponding system behavior.

9. Conclusion:

This project is built keeping in mind that it is to be used by user to analyze performance of the player and team. It can be further used by Cricket Training Academies to store performance of all the players in getting training in their academy by extending some functionality. According to the requirement user can add, update and delete data in the database. The required records can be easily viewed anytime time user wants in an instant. The main objective of this project is to save time, management of records efficiently.

10. Future Scope:

This project is specifically designed for the use of comparing statistics. It is especially useful for any sports enthusiastic and users of platform like DREAM11.

11. Github Project URL:

https://github.com/111903050/Cricket-Statistics-Management

12. References:

https://www.w3schools.com/REACT/DEFAULT.ASP

https://www.codecademy.com/learn/react-101