

The Cricinfo System

Cricinfo also known as ESPNcricinfo, is one of the leading cricket-related websites in the world. Website has a database including historical matches from the 18th century till the present. This website also provides articles & news about cricket. Moreover, people can search for matches, teams, players, tournaments etc. In short, this is a website for cricket lovers.

Expectations from the Interviewee

- How is information of the match updated?
- What is the process for adding commentary?
- At what intervals is the commentary added?
- What kind of stats does the system record?
- Does the system record the statistics of players & teams?
- What kind of people does our system need to store the information about, for example players, coaches etc?
- Will the Umpires data be stored in the system?
- Who will input data into the uricinfo system? What are the actors of the uricinfo system?
- How does the system keep track of different tournament?
- How are the tournament points & ranking stored in the system.

Requirement Collection

- R1: The system should be able to track the stat of all players, teams & matches.
- R2: The system should be able to track all scores or wickets that occurred for each ball. The system should also provide a live commentary for every ball.
- R3: The system should be able to keep track of all matches - Test, T20 & ODI matches
- R4: The system should be able to keep track of ongoing & previous tournaments. The system should also be able to show a points table for all teams participating in a tournament
- R5: The system should be able to show the result of all previous televised matches.

R6: All teams should select some players who will participate in tournament known as tournament squad

R7: For every match, the teams should be able to select 11 players to play on the field from the tournament squad known as the playing eleven.

R8: The admin of the system should be able to add tournaments, matches, teams, players & news to the system.

System



Primary Actors

↑ Admin
↓ Commentator

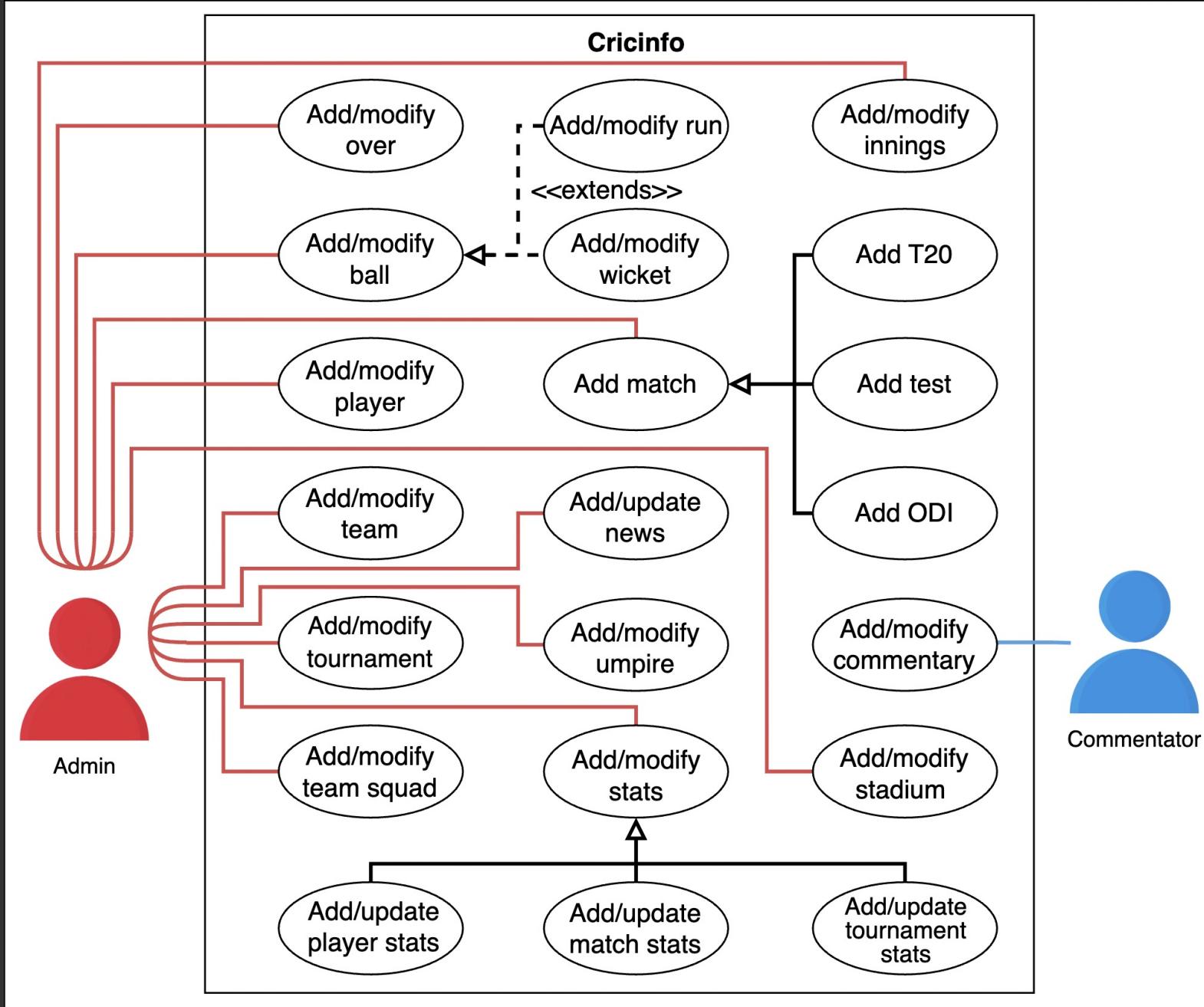
Secondary Actors

↳ no Secondary
Actors.

Admin	Commentator
Add/modify team	Add/modify commentary
Add/modify player	
Add/modify tournament	
Add/modify team squad	
Add/modify innings	
Add/modify over	
Add/modify ball	
Add match	
Add/update news	
Add/modify stadium	
Add/modify umpire	
Add/update stats	

← Associations

Use case diagram
of cricinfo



Class Diagram

1. Admin

Admin
<ul style="list-style-type: none"> - addPlayer() : bool - addTeam() : bool - addMatch() : bool - addTournament() : bool - addStats() : bool - addNews() : bool

2. Run, ball & wicket

Run
<ul style="list-style-type: none"> - totalRuns : int - type : RunType - scoredBy : Player

Ball
<ul style="list-style-type: none"> - balledBy : Player - playedBy: Player - type : BallType - runs : Run {list} - wicket : Wicket <p>+ addCommentary()</p>

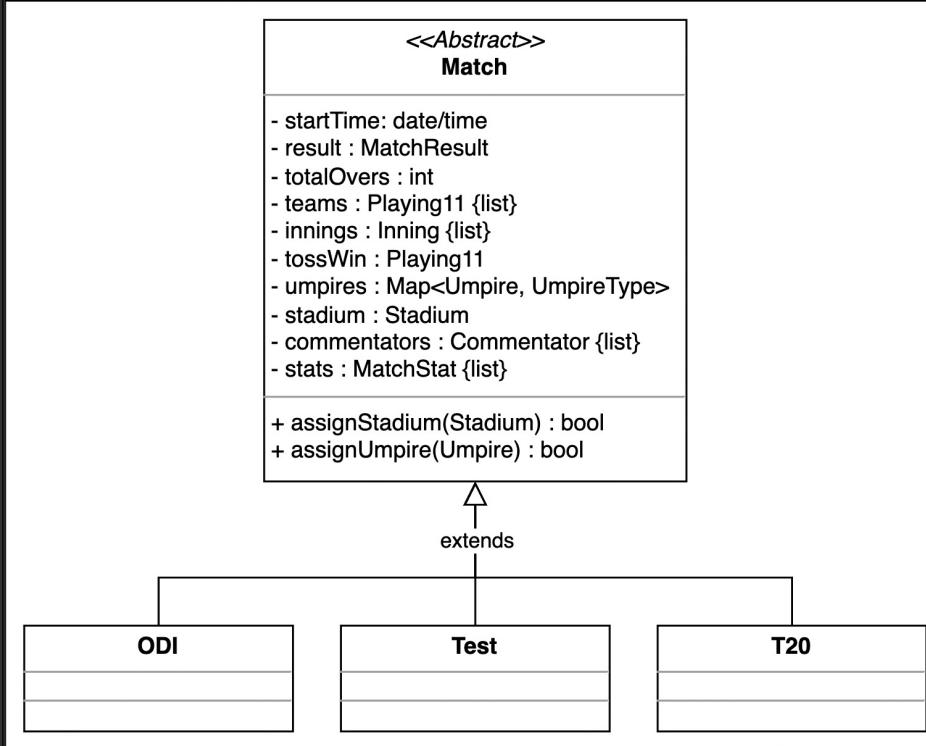
Wicket
<ul style="list-style-type: none"> - type : WicketType - playerOut: Player - balledBy : Player - caughtBy : Player - runoutBy : Player - stumpedBy : Player

3. Over & Innings

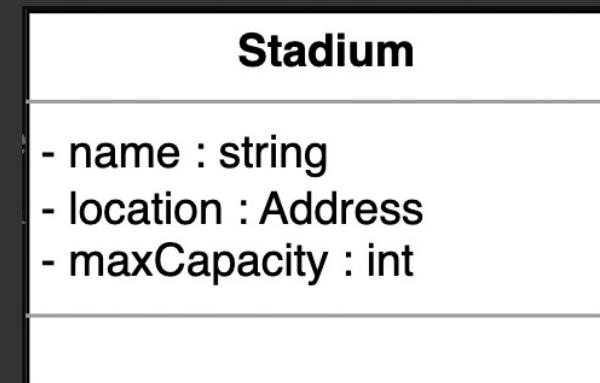
Over
<ul style="list-style-type: none"> - number : int - bowler : Player - totalScore: int - balls : Ball {list} <p>+ addBall() : bool</p>

Innings
<ul style="list-style-type: none"> - bowling : Playing11 - batting : Playing11 - startTime : date/time - endTime : date/time - totalScore : int - totalWickets : int - overs : Over {list} <p>+ addOver() : bool</p>

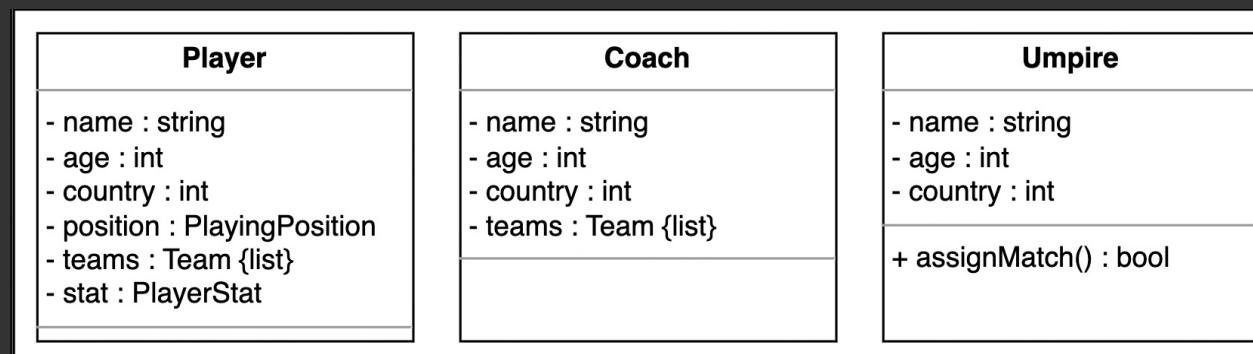
4. Match



5. Stadium



6. Player, Coach & Umpire



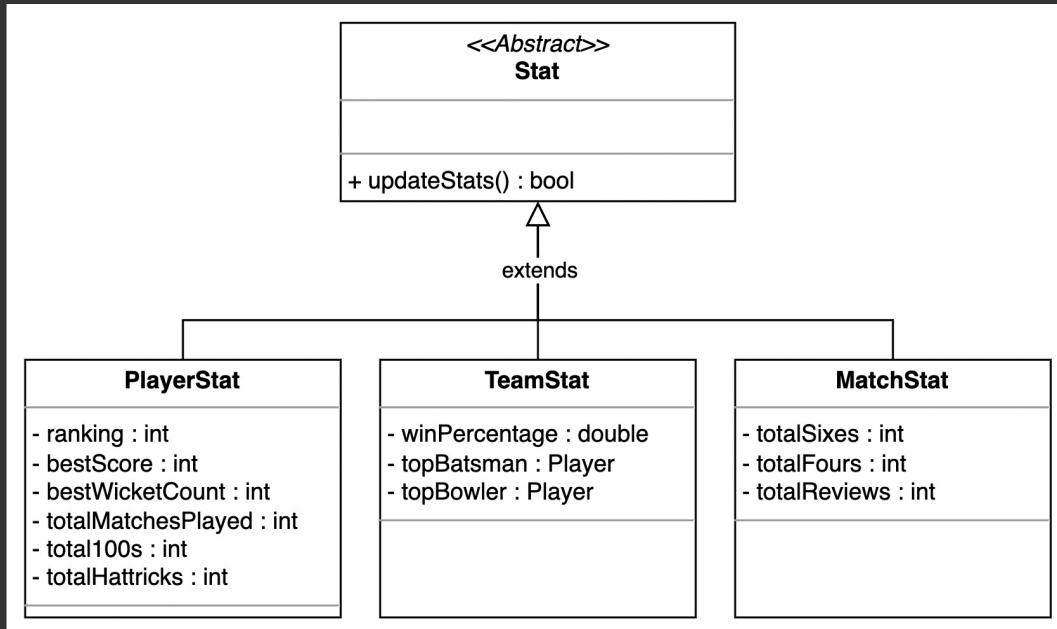
7. Team, tournament squad & playing Eleven

Team	TournamentSquad	Playing11
<ul style="list-style-type: none">- name : string- players : Player {list}- coach : Coach- news : News {list}- stats : TeamStat <ul style="list-style-type: none">+ addSquad(TournamentSquad) : bool+ addPlayer(Player) : bool+ addNews(News) : bool	<ul style="list-style-type: none">- players : Player {list}- tournament : Tournament- stats : TournamentStat <ul style="list-style-type: none">+ addPlayer(Player) : bool	<ul style="list-style-type: none">- players : Player {list} <ul style="list-style-type: none">+ addPlayer(Player) : bool

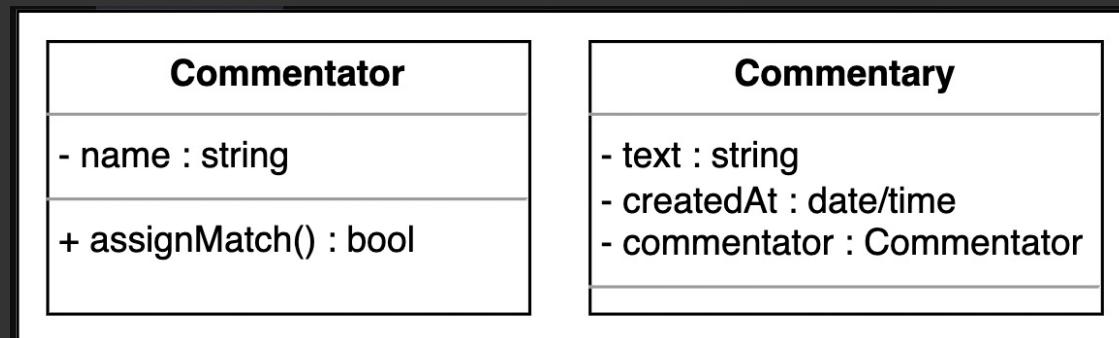
8. Tournament & points Table

Tournament	PointsTable
<ul style="list-style-type: none">- startDate : date/time- teams : TournamentSquad {list}- matches : Match {list}- points : PointsTable <ul style="list-style-type: none">+ addTeam() : bool+ addMatch() : bool	<ul style="list-style-type: none">- teamPoints : Map<string, float>- matchResults : Map<Team, MatchResult>- tournament : Tournament- lastUpdated : date/time

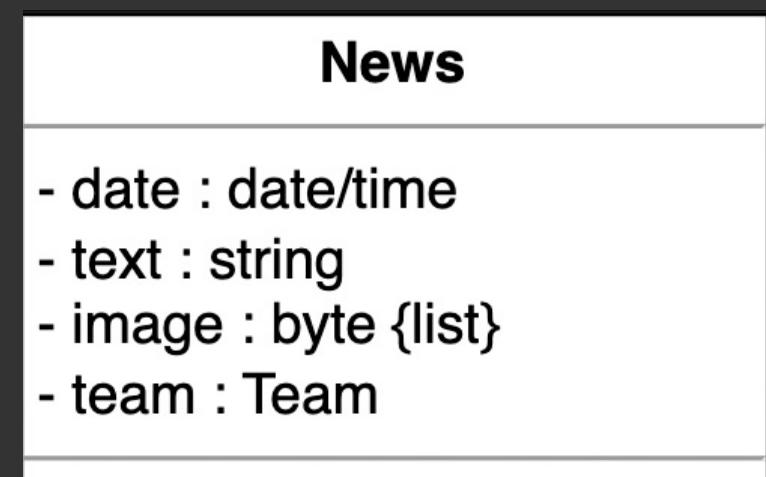
9. Stats



10. Commentator



11. News



Enumerations

<<Enumeration>>	RunType
Normal	
Four	
Six	
Wide	
LegBye	
Bye	
NoBall	
Overthrow	

<<Enumeration>>	MatchResult
Live	
BatFirstWin	
FieldFirstWin	
Draw	
Canceled	

<<Enumeration>>	WicketType
Bold	
Caught	
Stumped	
RunOut	
Lbw	
RetiredHurt	
HitWicket	
Obstruction	
Handled	

Address
- zipCode : int
- streetAddress : string
- city : string
- state : string
- country : string

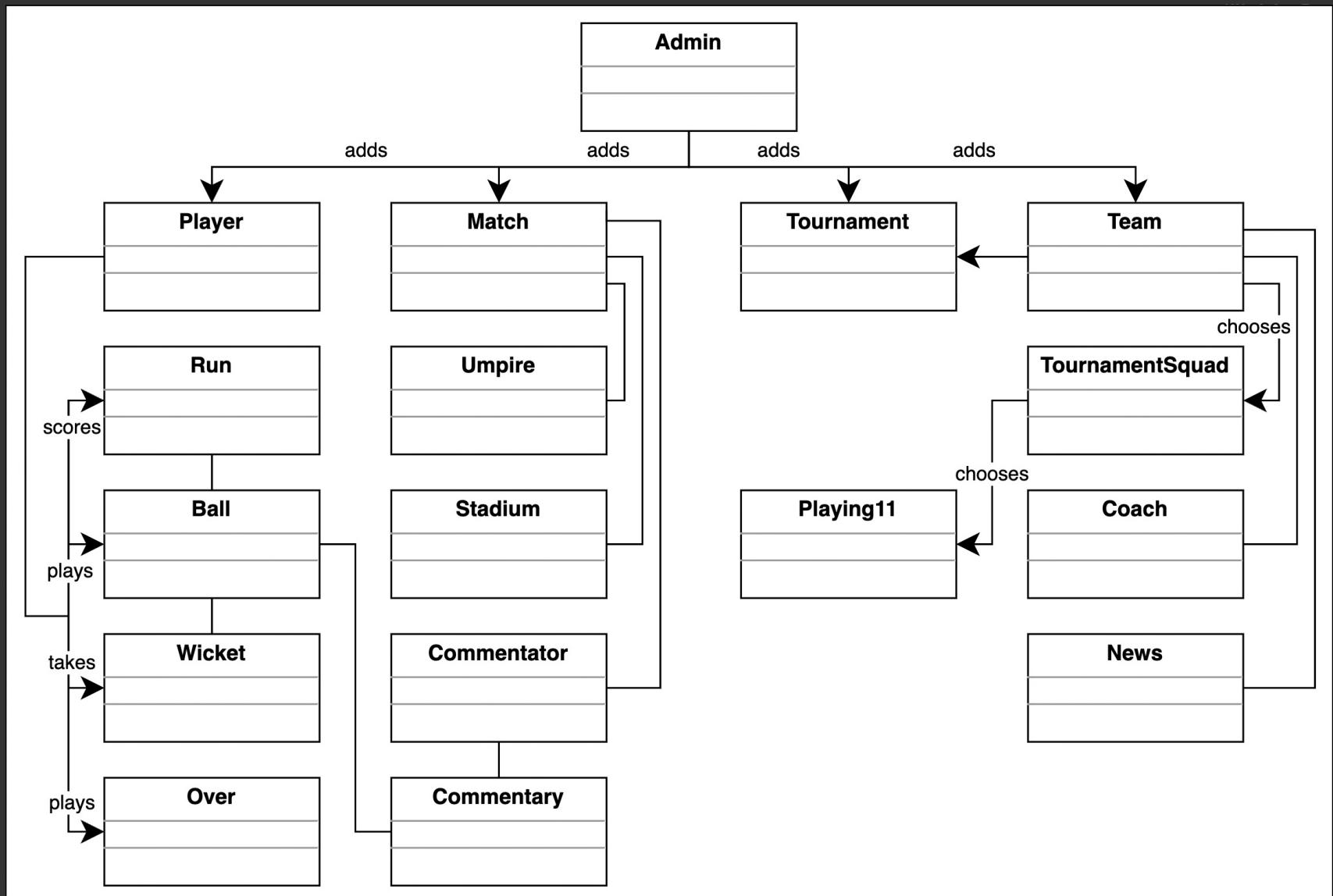
<<Enumeration>>	PlayingPosition
Batting	
Bowling	
AllRounder	

<<Enumeration>>	BallType
Normal	
Wide	
NoBall	
Wicket	

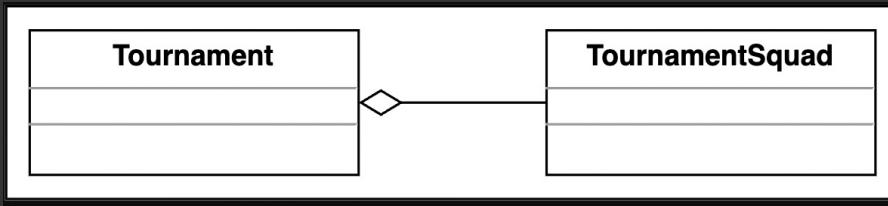
<<Enumeration>>	UmpireType
Field	
Reserved	
ThirdUmpire	

Relationships

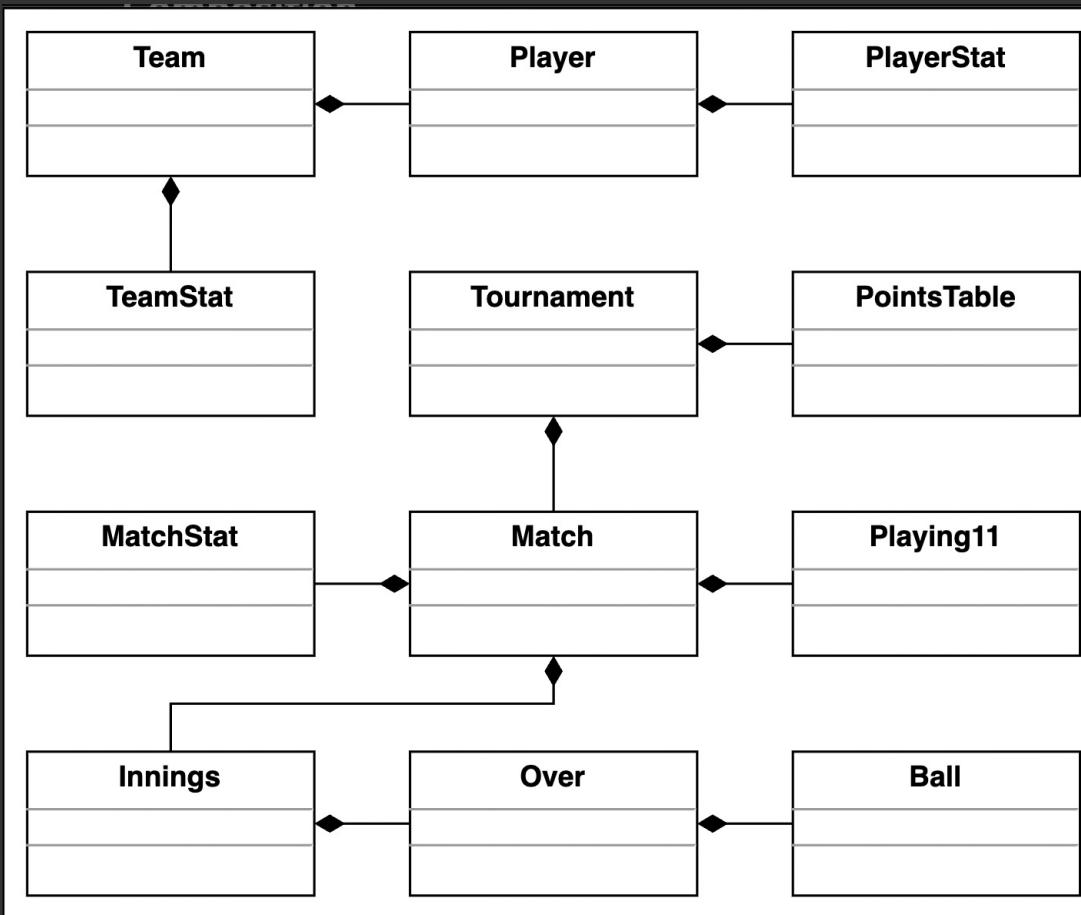
Two-way association

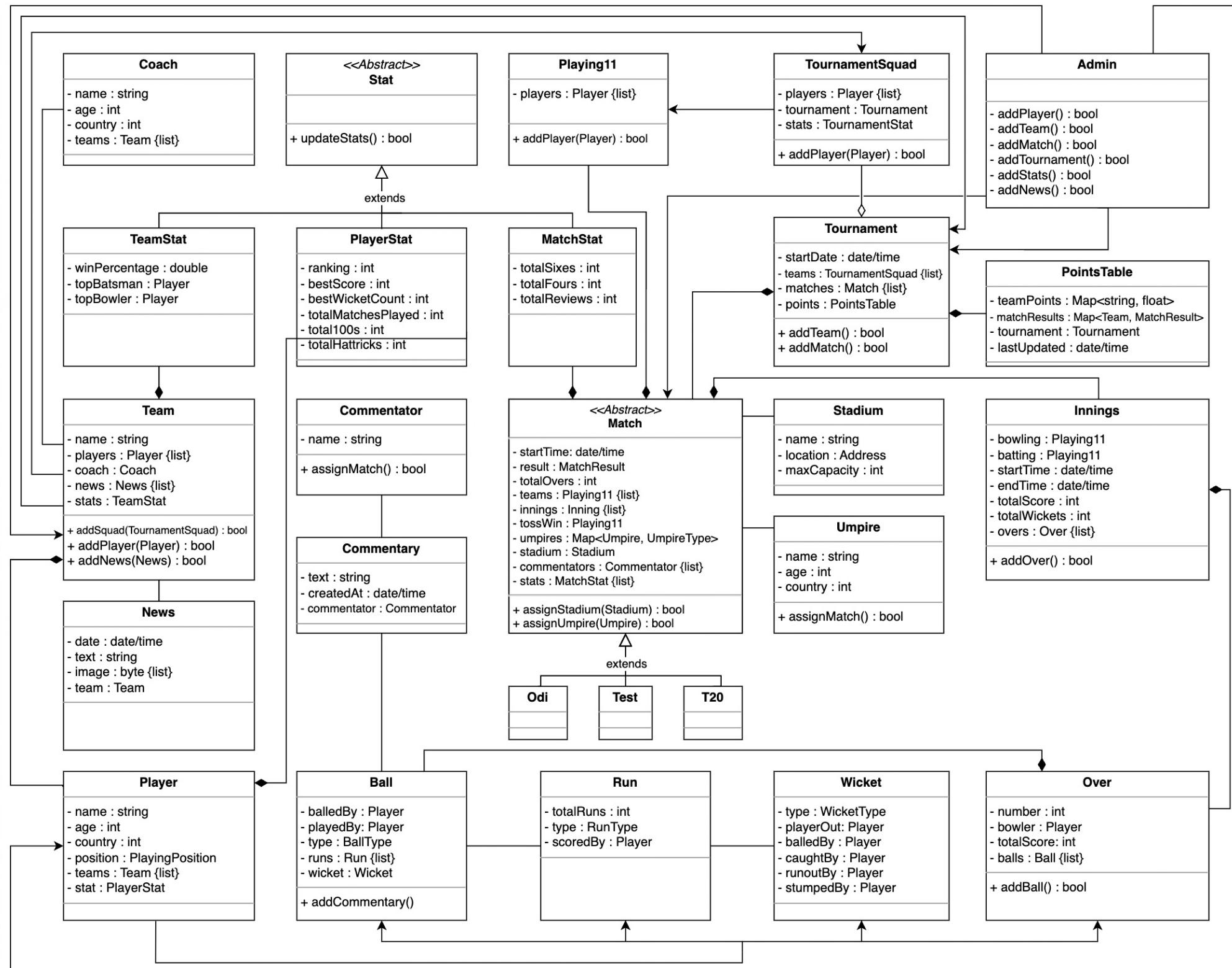


Aggregation

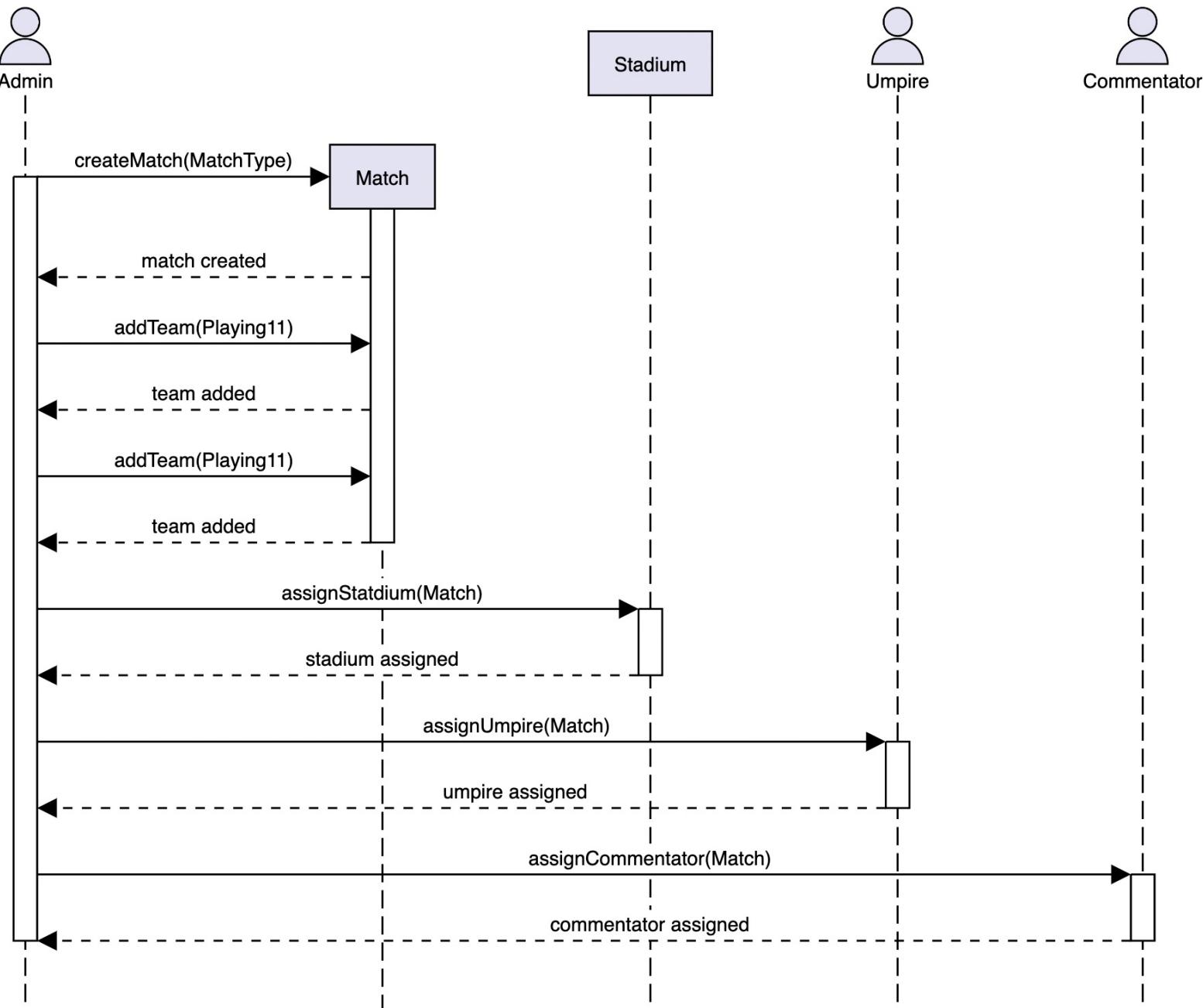


Composition

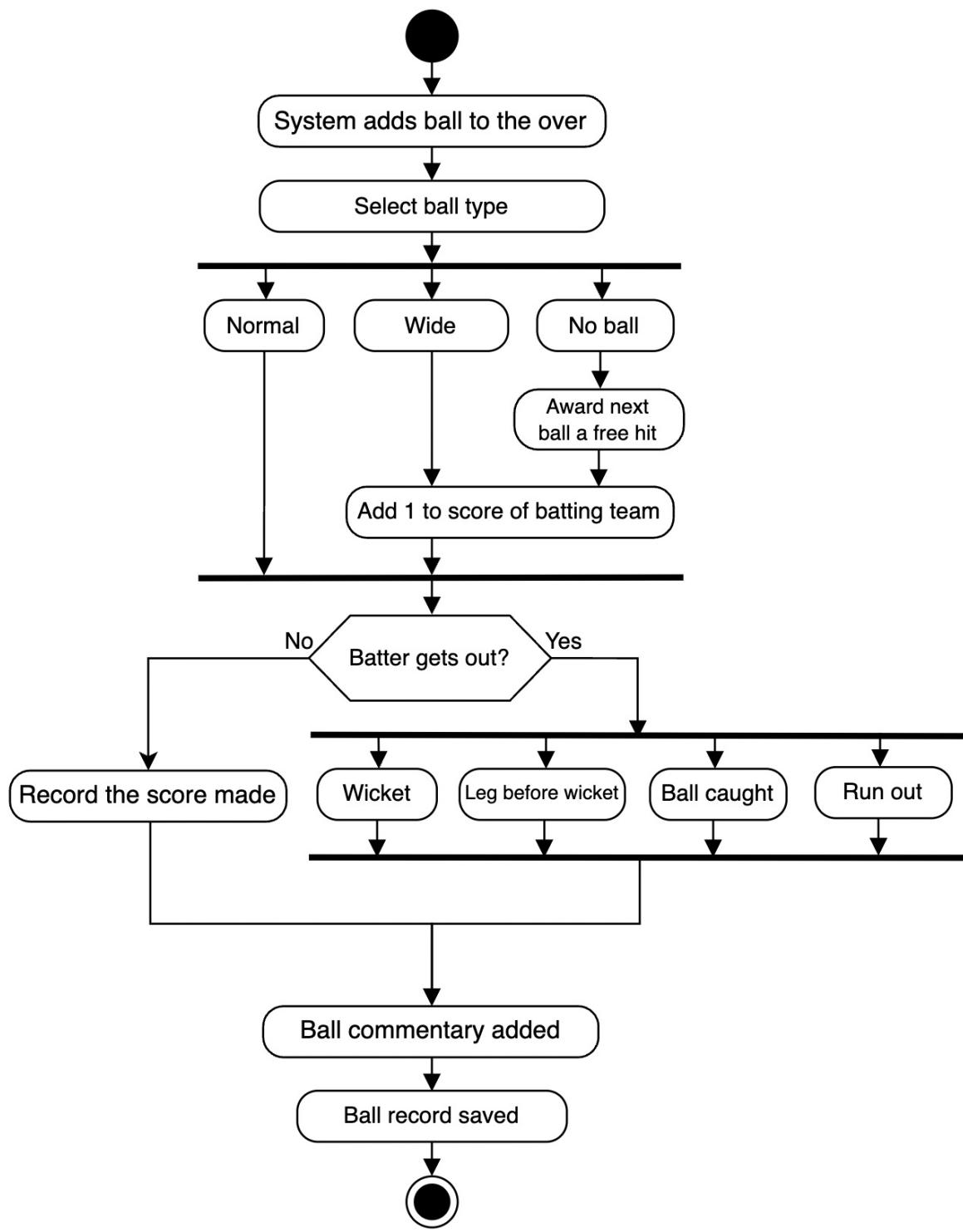




sd add match



Sequence diagram
of add match



Making a record of a ball.

1. Enumeration

```

public class Address {
    private int zipCode;
    private String streetAddress;
    private String city;
    private String state;
    private String country;
}

enum MatchResult {
    LIVE,
    BAT_FIRST_WIN,
    FIELD_FIRST_WIN,
    DRAW,
    CANCELED
}

enum UmpireType {
    FIELD,
    RESERVED,
    THIRD_UMPIRE
}

enum WicketType {
    BOLD,
    CAUGHT,
    STUMPED,
    RUN_OUT,
    LBW,
    RETIRED_HURT,
    HIT_WICKET,
    OBSTRUCTING,
    HANDLING
}

enum BallType {
    NORMAL,
    WIDE,
    NO_BALL,
    WICKET
}

enum RunType {
    NORMAL,
    FOUR,
    SIX,
    LEG_BYE,
    BYE,
    NO_BALL,
    OVERTHROW
}

enum PlayingPosition {
    BATTING,
    BOULING,
    ALL_ROUNDER
}

```

Model

2. Admin player coach umpire

```

public class Admin {
    public boolean addPlayer(Player player);
    public boolean addTeam(Team team);
    public boolean addMatch(Match match);
    public boolean addTournament(Tournament tournament);
    public boolean addStats(Stat stats);
    public boolean addNews(News news);
}

public class Player {
    private String name;
    private int age;
    private int country;
    private PlayerPosition position;
    private List<Team> teams;
    private PlayerStat stat;
}

public class Coach {
    private String name;
    private int age;
    private int country;
    private List<Team> teams;
}

public class Umpire {
    private String name;
    private int age;
    private int country;

    public boolean assignMatch(Match match);
}

```

3. Run, ball, wicket over & innings

```
public class Run {  
    private int totalRuns;  
    private RunType type;  
    private Player scoredBy;  
}  
  
public class Ball {  
    private Player balledBy;  
    private Player playedBy;  
    private BallType type;  
    private List<Run> runs;  
    private Wicket wicket;  
  
    public boolean addCommentary(Commentary commentary);  
}  
  
public class Wicket {  
    private WicketType type;  
    private Player playerOut;  
    private Player balledBy;  
    private Player caughtBy;  
    private Player runoutBy;  
    private Player stumpedBy;  
}  
  
public class Over {  
    private int number;  
    private Player bowler;  
    private int totalScore;  
    private List<Ball> balls;  
  
    public boolean addBall(Ball ball);  
}  
  
public class Innings {  
    private Playing11 bowling;  
    private Playing11 batting;  
    private Date startTime;  
    private Date endTime;  
    private int totalScores;  
    private int totalWickets;  
    private List<Over> overs;  
  
    public boolean addOver(Over over);  
}
```

4. Match

```
public abstract class Match {  
    private Date startTime;  
    private MatchResult result;  
    private int totalOvers;  
    private List<Playing11> teams;  
    private List<Innings> innings;  
    private Playing11 tossWin;  
    private Map<Umpire, UmpireType>  
umpires;  
    private Stadium stadium;  
    private List<Commentator> commentators;  
    private List<MatchStat> stats;  
  
    public abstract boolean  
assignStadium(Stadium stadium);  
    public abstract boolean  
assignUmpire(Umpire umpire);  
}  
  
public class T20 extends Match {  
    public boolean assignStadium(Stadium  
stadium);  
    public boolean assignUmpire(Umpire  
umpire);  
}  
  
public class Test extends Match {  
    public boolean assignStadium(Stadium  
stadium);  
    public boolean assignUmpire(Umpire  
umpire);  
}  
  
public class ODI extends Match {  
    public boolean assignStadium(Stadium  
stadium);  
    public boolean assignUmpire(Umpire  
umpire);  
}
```

Team, Tournament Squad, player 11

```
public class Team {  
    private String name;  
    private List<Player> players;  
    private Coach coach;  
    private List<News> news;  
    private TeamStat stats;  
  
    public boolean addSquad(TournamentSquad squad);  
    public boolean addPlayer(Player player);  
    public boolean addNews(News news);  
}  
  
public class TournamentSquad {  
    private List<Player> players;  
    private Tournament tournament;  
    private List<TournamentStat> stats;  
  
    public boolean addPlayer(Player player);  
}  
  
public class Playing11 {  
    private List<Player> players;  
  
    public boolean addPlayer(Player player);  
}
```

Tournaments, points Table & Stadium

```
public class Tournament {  
    private Date startDate;  
    private List<TournamentSquad> teams;  
    private List<Match> matches;  
    private PointsTable points;  
  
    public boolean addTeam(TournamentSquad team);  
    public boolean addMatch(Match match);  
}  
  
public class PointsTable {  
    private HashMap<String, float> teamPoints;  
    private HashMap<Team, MatchResult>  
matchResults;  
    private Tournament tournament;  
    private Date lastUpdated;  
}  
  
public class Stadium {  
    private String name;  
    private Address location;  
    private int maxCapacity;  
}
```

Commentator, Commentary, & News

```
public class Commentator {  
    private String name;  
  
    public boolean assignMatch(Match match);  
}  
  
public class Commentary {  
    private String text;  
    private Date createdAt;  
    private Commentator commentator;  
}  
  
public class News {  
    private Date date;  
    private String text;  
    private List<byte> image;  
    private Team team;  
}
```

Statistics

```
public abstract class Stat {  
    public abstract boolean updateStats();  
}  
  
public class PlayerStat extends Stat {  
    private int ranking;  
    private int bestScore;  
    private int bestWicketCount;  
    private int totalMatchesPlayed;  
    private int total100s;  
    private int totalHattricks;  
  
    public boolean updateStats();  
}  
  
public class MatchStat extends Stat {  
    private double winPercentage;  
    private Player topBatsman;  
    private Player topBowler;  
  
    public boolean updateStats();  
}  
  
public class TeamStat extends Stat {  
    private int totalSixes;  
    private int totalFours;  
    private int totalReviews;  
  
    public boolean updateStats();  
}
```