**1. Database use cases**

1. Show all players of a team
   1. Open the web browser
   2. Open the webpage
   3. Click on teams - to display all teams
   4. Click on particular team of interest
   5. Click on roster

SELECT Name FROM Player as p, Team as t WHERE t.club\_name = "Barcelona" and p.team\_ID = t.team\_ID

1. Teams standing for the season
   1. Open web browser
   2. Open the webpage
   3. Click on team standings

Select Club\_Name,Team\_Points from Team order by Team\_Points asc

1. View Player details
   1. Open the web browser
   2. Open the webpage
   3. Click on teams - to display all teams
   4. Click on particular team of interest
   5. Click on roster
   6. Click on particular player

SELECT \* from Player as p where p.Name = "Anurag Baral"

1. View Game details
   1. Open the web browser
   2. Open the webpage
   3. Click on teams - to display all teams
   4. Click on particular team of interest
   5. Pick a game of interest

Select \* from Game as g where team\_1\_id = "1" or team\_2\_id = "1"

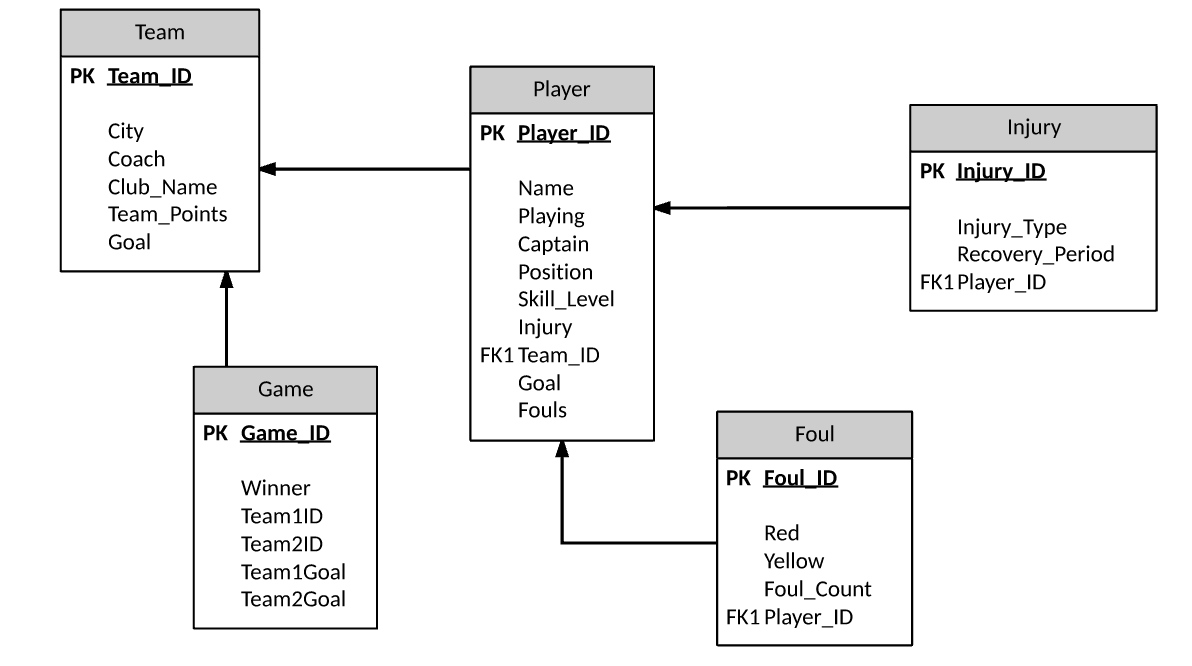
1. View all game schedules
   1. Open the web browser
   2. Open the webpage
   3. Click on “schedule”

Select \* from Game order by Time asc

1. Search for MVP of league
   1. Open the web browser
   2. Open the webpage
   3. Click on Rankings

Select Name, Goal from Player order by Goal desc limit 1

**2.1 Normalization: 1NF & 2NF**



**2.2 Dependencies and Normalization: 3NF**

**Table: Team**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Team\_ID | City | Coach | Club\_Name | Team\_Points | Goal |

Assumptions:

All attributes are dependent on Primary Key. No other attributes (except for primary key) will get you access to any other attributes in Team table. So in Team table, Club\_Name cannot give you the value for City, Coach, Team\_Points and Goal. All the attributes are the first level information of Team.

Logically it might seem possible that a Coach information can give you the information of the club\_name. But the assumption is Coach and Club\_name is both information given by the Team\_ID. Similarly with any attributes you can get to all the attributes only by using Team\_ID.

**Table: Game**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Game\_ID | Winner | Team1ID | Team2ID | Team1Goal | Team2Goal |

Assumptions:

All attributes are dependent on Game\_ID (the only primary key). For example, you cannot view Team1Goal just by accessing Winner or any other attributes. You have to relate the attribute(the one you want to view the data) with Game\_ID in order to access any other attributes.

Also the Team1Goal or Team2Goal is not the same as Goal of the Team table. So every attributes on this table is also the first level information of that particular game.

**Table: Player**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Player\_ID | Name | Playing | Captain | Position | Skill\_Level | Injury | Team\_ID | Goal | Fouls |

Assumptions:

Any Player’s information in a game can only be given via Player\_ID, which is the only primary key in the Player Table.

**Table: Foul**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Foul\_ID | Red | Yellow | Foul\_count | Player\_ID |

Assumptions:

Every other attribute is dependent on Foul\_id(Primary Key). You cannot access Foul information of any player if you don’t use Foul\_ID attribute.

**Table: Injury**

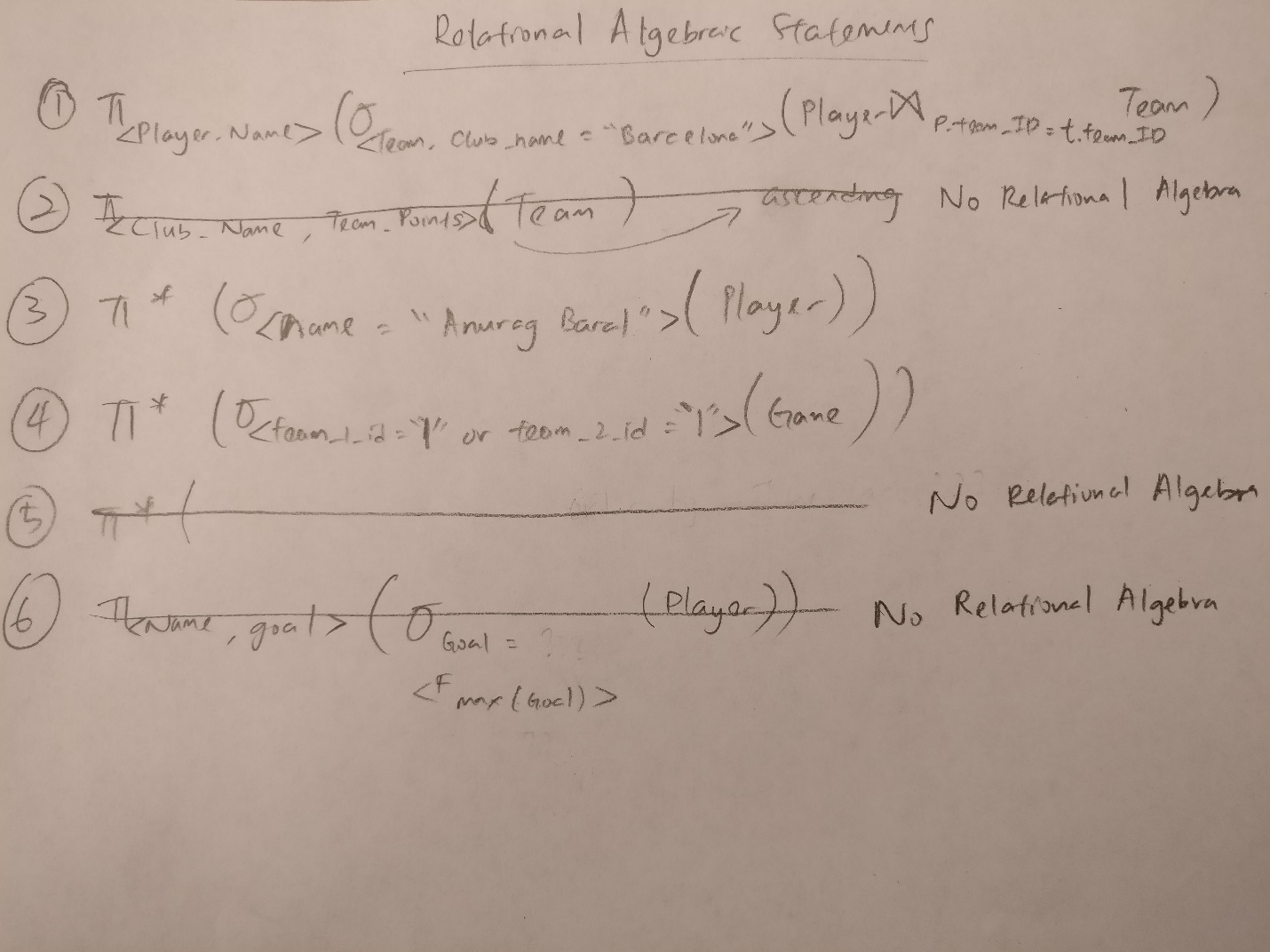
|  |  |  |  |
| --- | --- | --- | --- |
| Injury\_ID | Injury\_Type | Recovery\_Period | Player\_ID |

Assumptions:

If you want to retrieve an Injury information of a player, you can only do so by using the Primary key which is Injury\_Id. No other attributes can give you information of the other attribute.

**2.3 Relational Algebra**

Please view according to use case number sequence:

****

**2.4 SQL statements**

Please view according to use case number sequence:

1. SELECT Name FROM Player as p, Team as t WHERE t.club\_name = "Barcelona" and p.team\_ID = t.team\_ID

2. Select Club\_Name,Team\_Points from Team order by Team\_Points asc

3. SELECT \* from Player as p where p.Name = "Anurag Baral"

4. Select \* from Game as g where team\_1\_id = "1" or team\_2\_id = "1"

5. Select \* from Game order by Time asc

6. Select Name, Goal from Player order by Goal desc limit 1

**3.1 Description of MongoDB Design**

We created the MongoDB schema design in a way such that all the use cases could be retrieved. The first level information is games which contains array of single games. Inside games is the first level attributes like winner, team1goal, and others. Nested inside games is also another collection of teams. Teams contain all the attributes inside a team relation like city, coach, and the others mentioned on the schema. Inside teams is also a nested collection of players. There are many players with each having information such as name, playing, skill\_level. Inside players there are two nested collection injury and fouls which contains additional information.

The schema was created this way so that our use cases could be all facilitated! Look at the use cases for more information like attributes and relations!

**3.2 MongoDB database design**

Please refer to the file named “template.json” included in our team’s submission for better readability.

{

"games" : {

"game1" : {

"winner" : "Barcelona",

"team1goal" : 1,

"team2goal" : 3,

"teams" : {

"team1" : {

"city" : "Barcelona",

"coach" : "Anurag Baral",

"club\_name" : "Barcelona",

"team\_points" : 10,

"goal" : 20,

"players" : {

"player1" : {

"name" : "Lionel Messi" ,

"playing" : 1,

"captain" : 0,

"position" : "RW",

"skill\_level" : 92,

"injury" : {

"injury\_id" : {

"injury\_type" : "Shoulder Sprain",

"recovery\_period" : 30

}

},

"goal" : 2,

"fouls" : {

"red" : 0,

"yellow" : 0,

"foul\_count" : 1

}

},

"player2" : {

"name" : "Neymar" ,

"playing" : 1,

"captain" : 0,

"position" : "LW",

"skill\_level" : 90,

"injury" : {

"injury\_id" : {

"injury\_type" : "Leg Sprain",

"recovery\_period" : 3

}

},

"goal" : 2,

"fouls" : {

"red" : 0,

"yellow" : 0,

"foul\_count" : 1

}

}

}

},

"team2" : {

"city" : "London",

"coach" : "Julian Fung",

"club\_name" : "Arsenal",

"team\_points" : 8,

"goal" : 10,

"players" : {

"player1" : {

"name" : "Mezut Ozil" ,

"playing" : 1,

"captain" : 0,

"position" : "RW",

"skill\_level" : 87,

"injury" : {

"injury\_id" : {

"injury\_type" : "Shoulder Sprain",

"recovery\_period" : 2

}

},

"goal" : 3,

"fouls" : {

"red" : 0,

"yellow" : 0,

"foul\_count" : 1

}

},

"player2" : {

"name" : "Theo Walcott" ,

"playing" : 1,

"captain" : 0,

"position" : "LW",

"skill\_level" : 87,

"injury" : {

"injury\_id" : {

"injury\_type" : "Leg Sprain",

"recovery\_period" : 1

}

},

"goal" : 6,

"fouls" : {

"red" : 0,

"yellow" : 0,

"foul\_count" : 1

}

}

}

}

}

}

}

}

**4. Team member roles**

1. Sujan
   1. Brainstormed use cases
   2. Wrote step by step procedures for each use cases
   3. Discussed and participated in normalization of tables
   4. Database dependencies, SQL queries
   5. MongoDB design and schema
2. Harsha
   1. Brainstormed use cases
   2. Discussed and participated in normalization of tables
   3. Database dependencies, SQL queries
   4. Testing of SQL queries
   5. MongoDB design and schema
3. Anurag
   1. Brainstormed use cases
   2. Wrote step by step procedures for each use case
   3. Wrote step by step procedures for each use case
   4. Discussed and participated in normalization of tables
   5. Database dependencies, Relational algebraic statements in SQL
4. Julian
   1. Brainstormed use cases
   2. Wrote step by step procedures for each use case
   3. Visited Professor for approval of use cases
   4. Discussed and participated in normalization of tables
   5. Database dependencies, Relational algebraic statements in SQL
   6. Clarification of member roles