MLB Baseball Player and Team Final Report

Group 5: Nikhil Cherukuri, Daniel Guerrero, Nikita Potapov, Brittany Thum

https://danielguerrero.shinyapps.io/MLB_DBApp/

Table of Contents

Application Link

https://danielguerrero.shinyapps.io/MLB_DBApp/

Project Proposal

Brief introduction of the organization:

The Major League Baseball (MLB) organization is comprised of thirty teams. Each team has a general manager who controls all hiring and staffing transactions. Not only do they hire the coaching staff including the field manager (head coach), but they play a big roll in building the team player roster. The database we develop will contain the players and teams in the entire MLB organization and their game statistics. The app will use the database to provide the general managers a list of potential players to sign based on a parameter of statistics such as batting average, hits, etc. Together, the database and application will allow the general manager to build a stronger team and active roster to win more games. The team data will allow managers to understand their overall team performance and that of their competitors.

Scope of your project: What kinds of business functions are you going to cover?

The final database will group all statistics by player which will provide users the ability to learn about each player in the MLB from 20015 to 2021 and their performance scores. The database will display the details/statistics of each MLB player. This table will allow users to build a team roster based on performance statistics. Additionally, the app can be used as a tool for player management and currently reviewing team members, and the managers can decide which players their team should sell and buy/retain for future season transfers. The users can also few over team performance by viewing previous games. The team tables in the app can assist managers in understanding team performance against certain competitors and the trending scores. The visual analytics in the app also highlights how player count can influence game wins.

Goals of your project: Who are the intended end-users? What benefits does the organization obtain by implementing the system?

The intended users are all MLB General Managers or any user interested in learning about current baseball statistics. If there is a future team, owners and managers can use the app to determine which players to recruit and how they may play competitively against other teams in the MLB. By implementing analytics into the drafting process, the general manager will have access to a much wider array of information about any given player's performance and will be able to determine a player's best fit in the team. By having a user-friendly interface, the general manager will be able to start making decisions almost immediately after applying this solution/application, and the process of doing so will be more efficient. The result is the team's improved performance which translates into higher revenues by attracting bigger sponsors/advertisers, creating bigger viewership for games, and by increasing ticket sales.

Design and E-R Diagram

(See appendix for full E-R Diagram)

The E-R Diagram consists of four entities:

Team: The 30 MLB teams which includes the *Team Name* and *Players*

Player: Baseball player in the MLB which includes the *Player ID, Player Name, Team*, and *Player Current Form*

Game: Game is played between two MLB teams and includes the *Player ID, Date, Team, Opponent, Date, Team,* and *Result*

Performance Indicators: This entity contains the player statistics which includes *Player ID, Date, Date, Player ID, Plate Appearances, At Bats, Runs, Hits, Doubles, Triples, Home Runs, RBIs, Base on Balls, Strikeouts, Stolen Bases*.

The E-R Diagram is designed with these four entities because our app will have pages for player and team data. Additionally, the design helps explain the relationship between the tables just like how a Player is part of a Team, plays in Games, and has Performance Indicators. The design purposefully shows this so that it can also be used in an app when visualizing the data.

Transformation:

The raw data consisted of the MLB player performances from 1901 to 2021. As the dataset consisted of millions of values, we decreased its size to make the application simpler by taking the values only from 2015 to 2021. The data contained the date, team, opponent, player ID, player scores, etc. We divided this data into four different aspects based on Team info, Player info, Game info and Player Indicators. As the raw data was also in the tables format, it was easier to just create new data by selecting the required columns from the original one.

Normalization:

To connect the entities, we had common fields between connecting tables like the Player ID, Date, and Team. These were designed as the primary and the foreign keys to connect the tables in an easy manner. These relationships are outlined in the appendix with the E-R Diagram. We made sure to remove the redundancies other than the connecting fields, making access simpler and thereby normalizing the tables. Then we also made sure that the tables don't have any transitive dependencies which resulted in the final normal form.

Shiny User Interface

(See appendix for screenshots of the App)

User Interface:

The Shiny app was designed to perform CRUD operations, and the user interface has been built to be accessed by MLB team managers. This application simplifies the task of analyzing the player and team performance to build a stronger team for upcoming games. The

app has seven pages beginning with an introduction page, two pages with a search function (one for players and one for teams), three pages to add, change/update, and remove game data, and a final page for analysis and visualization.

User Workflow:

The introduction page of the app welcomes the general managers and provides a brief description of the app functions. There is also a team list with their abbreviations that will be used for the search, add, update, and remove functions.

The second and third page have search functions and data tables with players and team information respectively. The second page is dedicated to the players. The user can search by team or leave it blank, and a data table appears with a list of player name and their performance statistics: runs, hits, at bats, and strikeouts. If the user searches by team, then the only players listed in the table are ones who have played for that team. The user can also use the rank buttons on the table to see which player has the highest number of runs, etc.

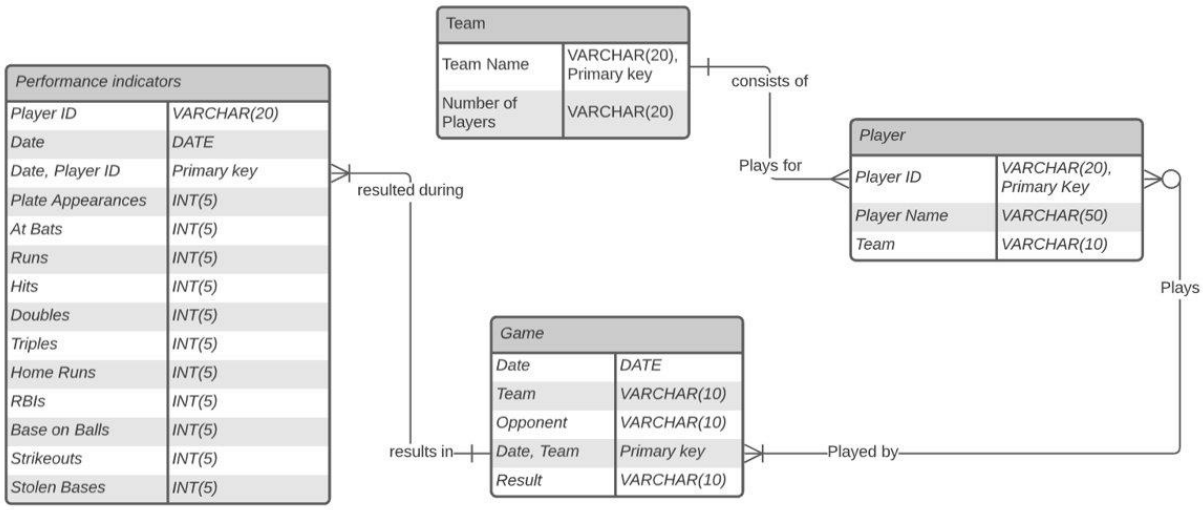
The third page has the search function for the team games. Similarly, to page 2, the user can search by team or leave it blank to view all games from 2015 to 2021. The table lists the games out by date, team, opponent, and result. For example, if a user searched for the Chicago Cubs, they would only see the games the Cubs have played. If the user ranks the table by date in descending order, they will be able to see the trailing games and their scores. This may help a user gain trend information on team game performance.

The fourth, fifth, and sixth page have the add, update, and remove functions. A user will be able to add details of a new game into the app. This is useful to keep the app updated in real time. The user can add the date of the game, the team, opponent, and result. To change game information on the fifth page, the user needs to input the date, team, and new score. This may be useful in cases of cheating or late minute organization changes to a final score. Lastly, the user is able to remove a game on page six. Only the date and team are needed for inputs. This function will allow managers to remove games that they feel may be outliers for their analysis purposes.

The last page contains the analysis of the data. Once a user clicks display, two bar graphs appear, and they visualize the total number of players by team vs the total wins. The visualization provides a competitive analysis of all the teams in the MLB. Managers can determine how number of players over time affects total wins. There are a few outliers like the LA Dodgers, but overall, the more players (indicating a lot of trades, etc.), does not equate to more wins.

Appendix

1) Relationships



A team consists of many players.

Players play for one team.

Players play one-to-many games.

Game is played by multiple players.

Game results in one-to-many performance indicators.

The Performance indicators are resulted from single game.

2) Data Dictionary

Entities:

Team: The 30 MLB teams which includes the *Team Name* and *Players*

Player: Baseball player in the MLB which includes the *Player ID*, *Player Name*, *Team*, and *Player Current Form*

Game: Game is played between two MLB teams and includes the *Player ID, Date*, *Team, Opponent*, *Date, Team*, and *Result*

Performance Indicators: This entity contains the player statistics which includes *Player ID, Date, Date, Player ID, Plate Appearances, At Bats, Runs, Hits, Doubles, Triples, Home Runs, RBIs, Base on Balls, Strikeouts, Stolen Bases*.

Teams Names as shown in the App:

Arizona Diamondbacks - ARI
Atlanta Braves - ATL
Baltimore Orioles - BAL
Boston Red Sox - BOS
Chicago Cubs - CHC
Chicago White Sox - CHW
Cincinnati Reds - CIN
Cleveland Guardians - CLE
Colorado Rockies - COL
Detriot Tigers - DET
Houston Astros - HOU
Kansas City Royals - KRC
Los Angeles Angels - LAA
Los Angeles Dodgers - LAD
Miami Marlins - MIA
Milwaukee Brewers - MIL
Minnesota Twins - MIN
New York Mets - NYM
New York Yankees - NYY
Oakland Athletics - OAK
Philadelphia Phillies - PHI
Pittsburgh Pirates - PIT
San Diego Padres - SDP
Seattle Mariners - SEA
San Francisco Giants - SFG
St Louis Cardinals - STL
Tampa Bay Rays - TBR
Texas Rangers - TEX
Toronto Blue Jays - TOR
Washington Nationals - WSN

Performance Metrics as shown in the App:

(Definitions via MLB.com)

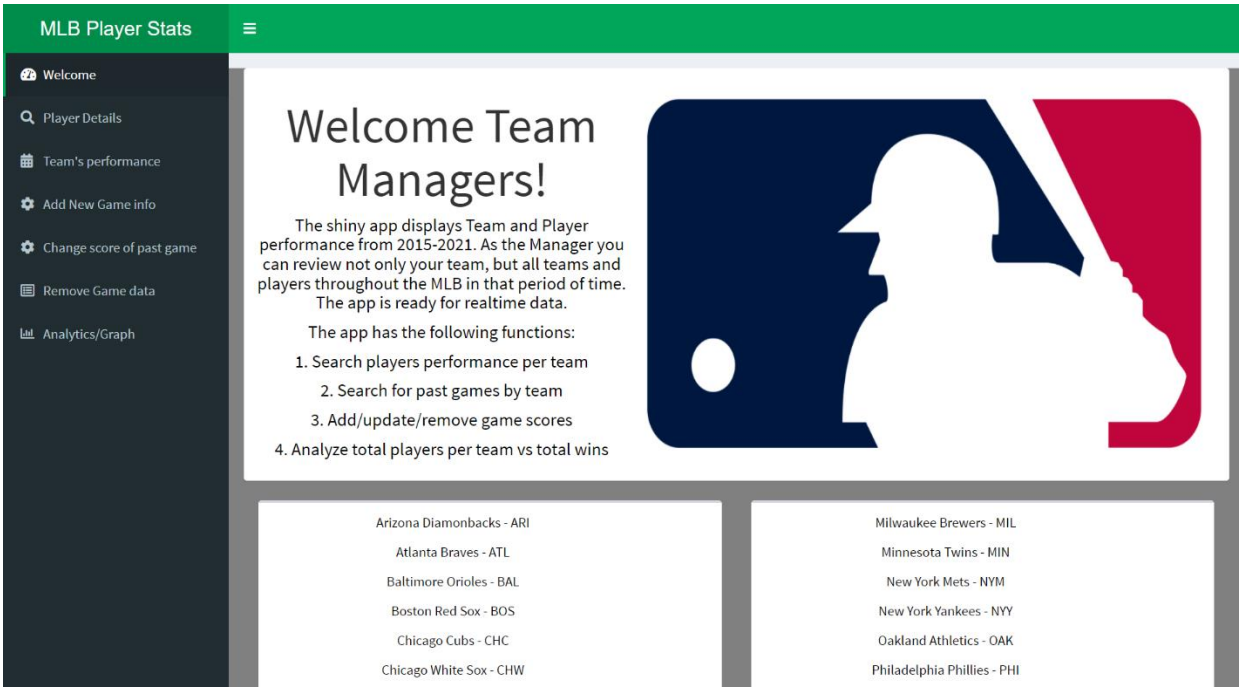
Runs: When a player crosses home plate to score a run.

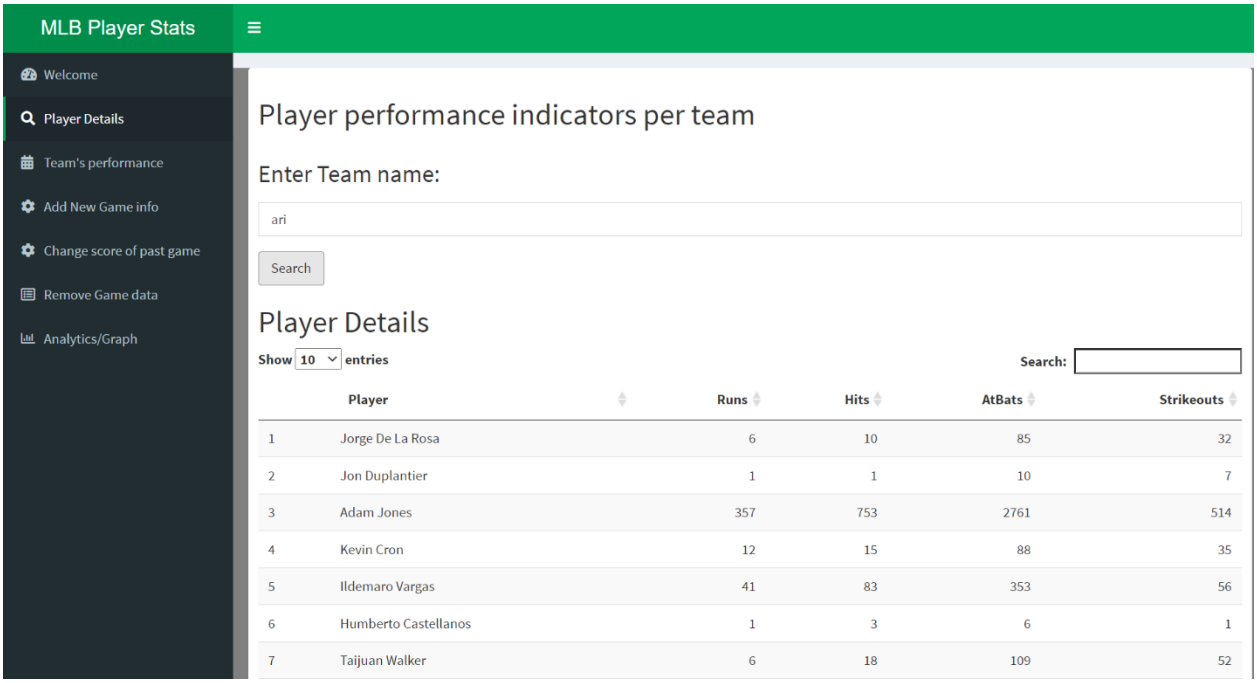
Hits: When a batter strikes the ball and reaches base.

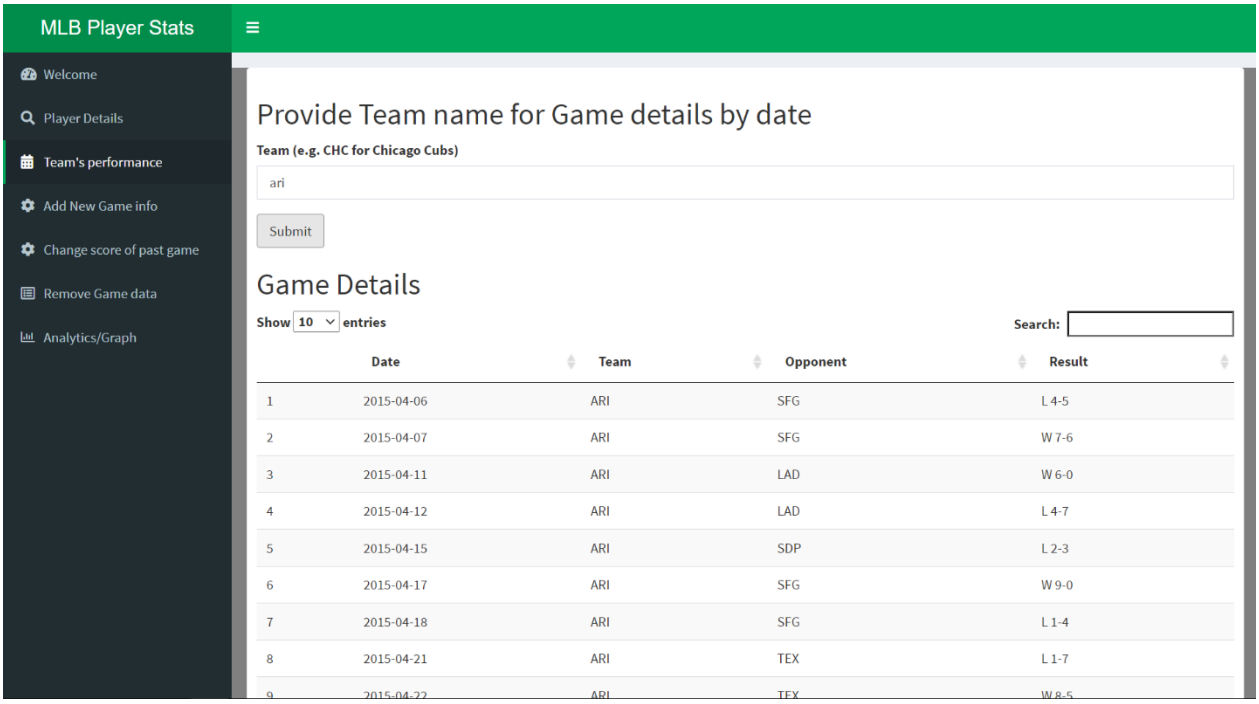
At Bats: When a batter reaches base via a field players choice, hit, or error. The denominator when calculating the batting average.

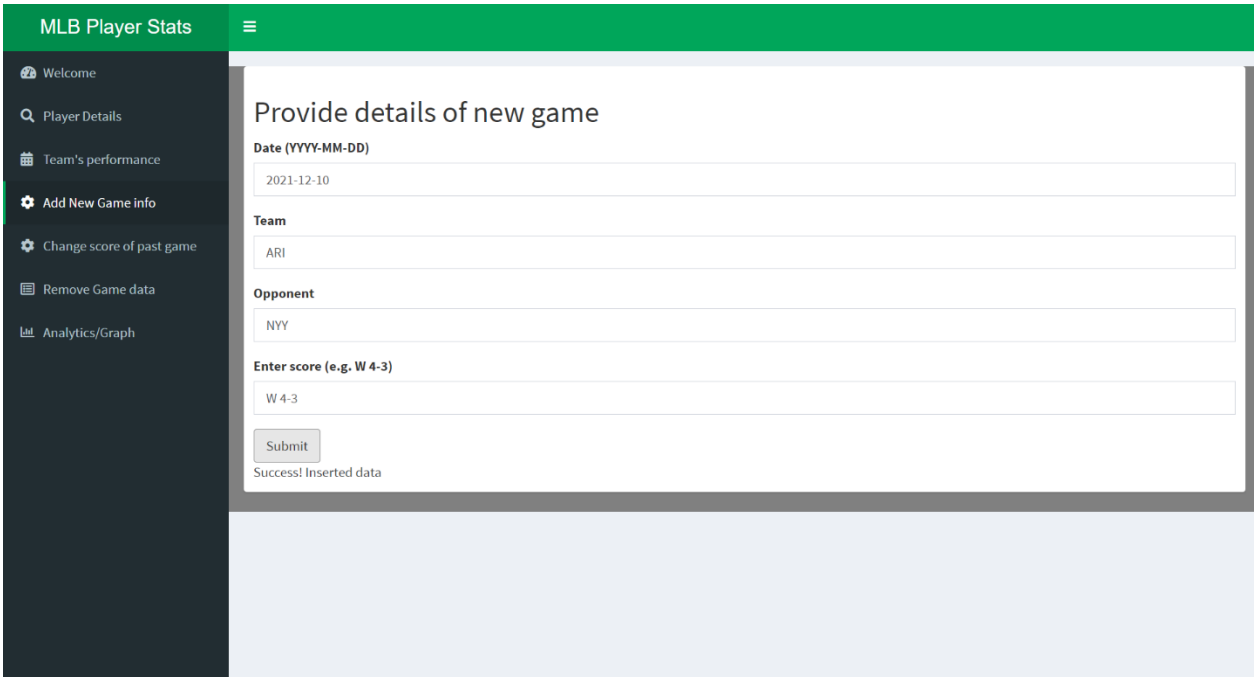
Strikeouts: When a batter receives three strikes.

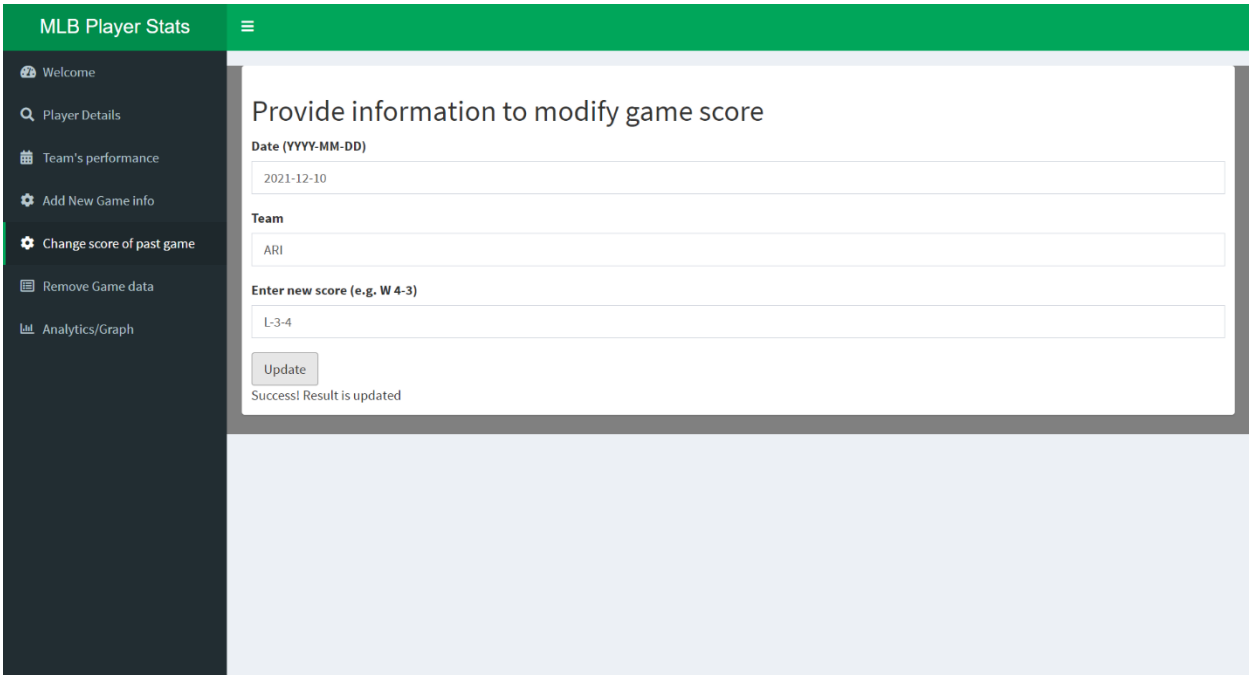
3) App Screenshots

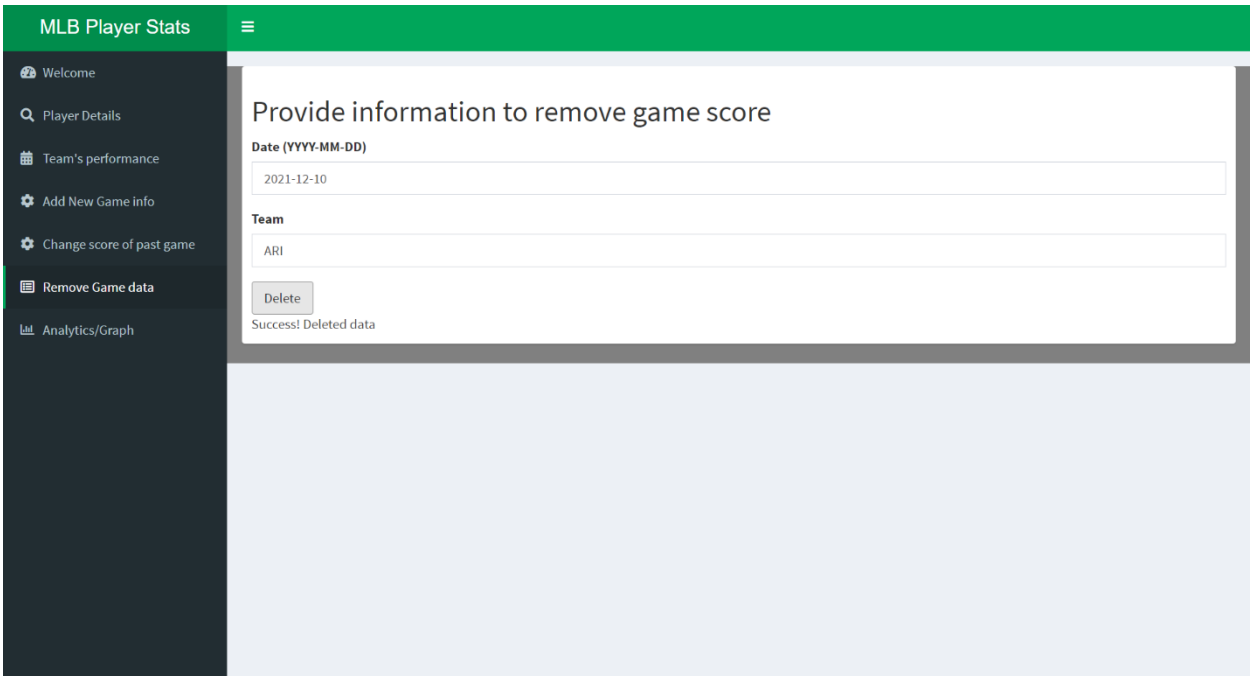


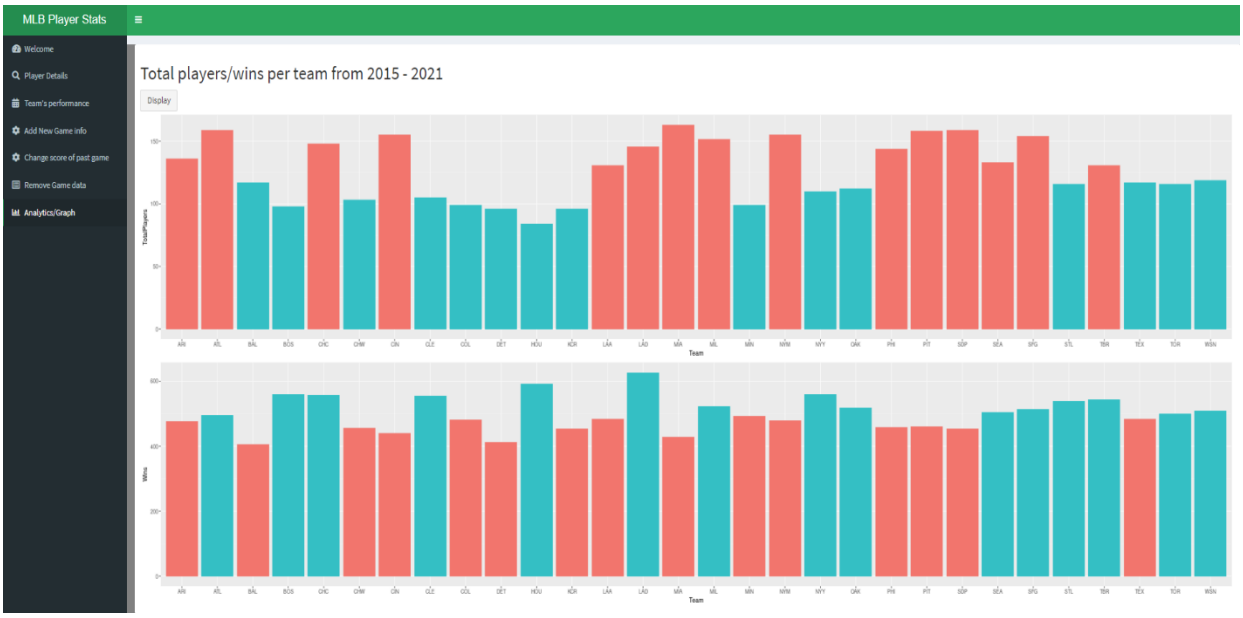



MLB Player Stats
Welcome
Player Details
Team's performance
Add New Game info
Change score of past game
Remove Game data
Analytics/Graph
Provide Team name for Game details by date
Team (e.g. CHC for Chicago Cubs)
ari
Submit
Game Details
Show 10 entries
Search:
Date
Team
Opponent
Result
1 2015-04-06 ARI SFG L 4-5
2 2015-04-07 ARI SFG W 7-6
3 2015-04-11 ARI LAD W 6-0
4 2015-04-12 ARI LAD L 4-7
5 2015-04-15 ARI SDP L 2-3
6 2015-04-17 ARI SFG W 9-0
7 2015-04-18 ARI SFG L 1-4
8 2015-04-21 ARI TEX L 1-7
9 2015-04-22 ARI TEX W 8-5


MLB Player Stats
Welcome
Player Details
Team's performance
Add New Game info
Change score of past game
Remove Game data
Analytics/Graph
Provide details of new game
Date (YYYY-MM-DD)
2021-12-10
Team
ARI
Opponent
NYY
Enter score (e.g. W 4-3)
W 4-3
Submit
Success! Inserted data


MLB Player Stats
Welcome
Player Details
Team's performance
Add New Game info
Change score of past game
Remove Game data
Analytics/Graph
Provide information to modify game score
Date (YYYY-MM-DD)
2021-12-10
Team
ARI
Enter new score (e.g. W 4-3)
L-3-4
Update
Success! Result is updated


MLB Player Stats
Welcome
Player Details
Team's performance
Add New Game info
Change score of past game
Remove Game data
Analytics/Graph
Provide information to remove game score
Date (YYYY-MM-DD)
2021-12-10
Team
ARI
Delete
Success! Deleted data


MLB Player Stats
Welcome
Player Details
Team's performance
Add New Game info
Change score of past game
Remove Game data
Analytics/Graph
Total players/wins per team from 2015 - 2021
Display
TotalPlayers
Team
Wins
Team