BBDB (Basketball Database)

Our project, BBDB (Basketball Database), takes sports data analytics to the next level. We're focusing on basketball, which has an incredible amount of data. It's a web app that makes it easy for the user to get whatever data they want, for any year, game, team, and even player.

The search feature will have many different parameters that can be customized to provide all the information the user needs. The added visualizations and other features help make the data easier to understand and compare, and other players/teams/etc with similar stats can be suggested. BBDB expands on what can be done with all of the basketball data available nowadays.

One application of our project is performance analysis for teams and coaches to use. By creating a visual representation and a textual representation for a specific statistic(s) for a player or team, it is easy for team analysts to aggregate data and make improvements to their team. With analytics becoming ever more prevalent in the NBA world, there is high demand for a tool like this. There is a lot of NBA raw data that exists right now, but there isn't a tool to aggregate it and present it in a more conclusive way.

A good creative component to improve the functionality of our application is a related search feature. So when a user looks up/queries a stat or player that they're interested in, along with the normal output, we will list some related searches that they may also be interested in based on what they just searched. For example, if they search up Lebron James 2021 season averages, some related searches may be Lebron James 2020 season averages. This will be done through nlp tokenization and distance functions. Another creative component can be various types of graph visualizations that provide the user with more concise data and allow them to make conclusions easier. We plan to make a dynamic and interactive graph, where the graph changes based on the query from the user. The graph will display various statistics through a time series graph based on the user's input. The general purpose of this component will be to provide the user with a visual representation of statistics to complement the textual information from the queries. The challenging component of this will be writing methods to determine which type of graph is most fitting for the query. For example, if we are looking at average win shares for a player over their career, a line graph is recommended since it accurately displays time series data and we are able to compare it to players similar to their caliber. Additionally, for searches that meet the criteria, we will add visualization of shot charts representing distance and results of shots.

Our application is useful because it allows individuals interested in NBA statistics and NBA team analysts to see trends in data both in text and graphically. Users can look up specific players or teams, and check various statistics. There are many input boxes for the user to choose players, teams, and statistics. Based on the input, it will display the appropriate statistic, a graphical representation, and potential future queries the user may want to input. This is our advanced component, as giving future recommendations and other player/team statistics involve potential machine learning or generative AI. There is a similar website called statmuse, which has the user input a singular query and there is a statistic output. What differs between our project and stat muse is we have graphical representations along with statistics. We also provide additional content in the form of future potential queries and more information output.

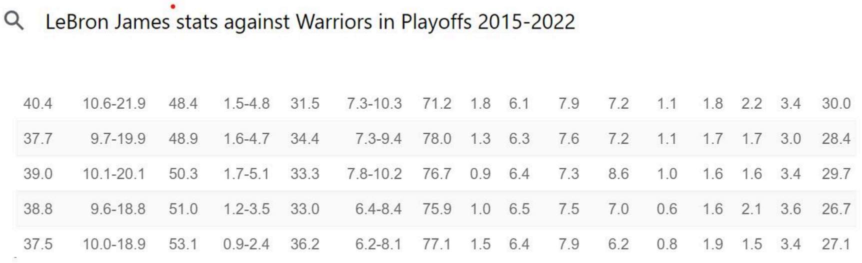
Our data sources will be basketball reference and nba.com. Basketball reference data can be retrieved using python web scraping packages, while nba.com can be directly accessed through its public API. The format of the data retrieved will be returned in a JSON format, which can then be saved as a csv file. The type of basketball data we will retrieve will result in multiple different types of tables of varying degree. For example, one table could be for game data, which would have columns like teams playing, date, final score etc. Similarly, there could be a player specific table describing career stats or even per game player stats. In general, there will be a lot of rows in the tables along with a high cardinality. Additionally, we will be using an NBA stats API that gives us access to more specific data, as detailed as the result of every play in the game.

We will have functionality for all the aspects of CRUD. We will be using a lot of different data to fulfill user queries, since they might want multiple details from a player's stats throughout an entire season, for example. The user will be able to add and update information such as their favorite players and teams, so the homepage can provide updates on the latest stats from their recent games. This will be flexible to allow the user to specify what stats they want to have most conveniently. We will have a secure sign-in feature that requires a username and password. The core functionality revolves around a search system that will provide any statistics you can think of for all of the basketball teams in the NBA. These can be player level, team level, season level, anything.

Responsibilities:

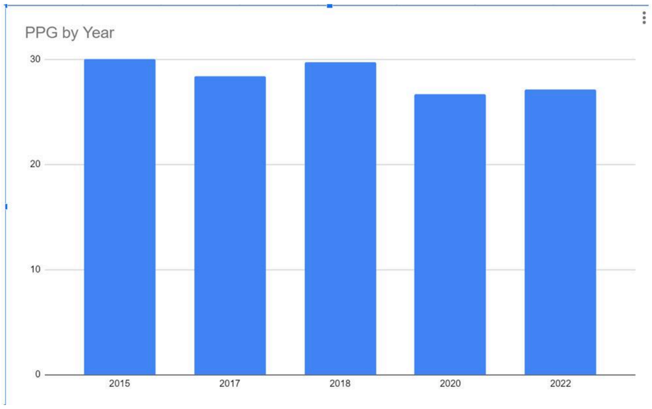
- Frontend: None of us are frontend experts, so we can all support each other and figure it out.
- Backend:
  - Michael - Converting the user search parameters into the correct SQL queries.
  - Max - Create visualizations for each query including graphs, plots, and charts
  - Brian - Source data from basketball reference and nba.com api and organize information into database tables. Also related searches.
  - Abhi - Run algorithm based on players present in the games stats are pulled from to give recommendations of other players stats to look at

Nicknames: King James,
The Chosen One, LeSunshine



LeBron James stats against Warriors in Playoffs 2015-2022

40.4	10.6-21.9	48.4	1.5-4.8	31.5	7.3-10.3	71.2	1.8	6.1	7.9	7.2	1.1	1.8	2.2	3.4	30.0
37.7	9.7-19.9	48.9	1.6-4.7	34.4	7.3-9.4	78.0	1.3	6.3	7.6	7.2	1.1	1.7	1.7	3.0	28.4
39.0	10.1-20.1	50.3	1.7-5.1	33.3	7.8-10.2	76.7	0.9	6.4	7.3	8.6	1.0	1.6	1.6	3.4	29.7
38.8	9.6-18.8	51.0	1.2-3.5	33.0	6.4-8.4	75.9	1.0	6.5	7.5	7.0	0.6	1.6	2.1	3.6	26.7
37.5	10.0-18.9	53.1	0.9-2.4	36.2	6.2-8.1	77.1	1.5	6.4	7.9	6.2	0.8	1.9	1.5	3.4	27.1



Check out Stats for:

Stephen Curry

Kevin Durant

Kyrie Irving

PPG

RPG

APG

PER