Player Comparator

Process Book

Overview and Motivation:

The National Basketball Association (NBA) is one of the most popular sports leagues in the world. Comparing basketball players of the NBA is a common pastime among basketball fans, though they usually depend on faulty memory instead of pre-established data. For example, Michael Jordan is consistently compared to Lebron James when discussing “who is the best player of all time?” Some would argue Michael Jordan is better because he won six NBA championships, while others would argue Lebron James is better because he has better stats in more categories, such as assists and rebounds. Therefore, in order to end debates like this, we created the NBA Player Comparator to easily distinguish who the best players are by comparing multiple stats in an easily digestible manner.

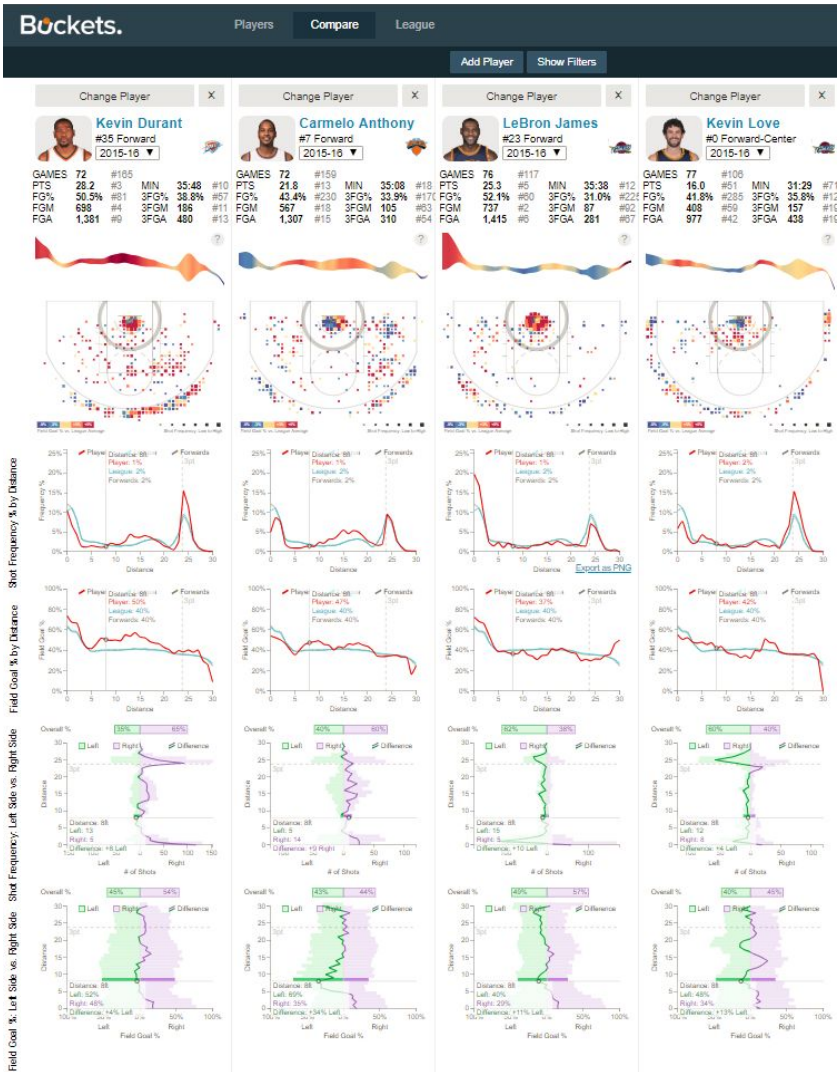
We originally intended this to be a tool used primarily by NBA fans who have some general knowledge of basketball. However, we realized we could make this more accessible to the general audience by including definitions of the stats that would explain certain aspects of the game. For example, hovering over the “APG” text gives the definition of what “Assists Per Game” means, as well as the definition of a single assist. We direct users to these hover features by including “tooltips” in the form of question marks that explain the features of the visualization and how to access more information.

Our main goal was to provide an easy-to-use tool that allows users to explore their comparisons between the greats of basketball. We intended to give the user a general idea of how good a player is with a chart that shows the stats of all their seasons in the “Average Stats for Each Season” chart, and then how effective that player is in contributing to their team’s winning record by showing how far they got in the playoffs in the “Playoff Results for Each Season” chart. We then allow the user to explore how the player progressed throughout each season in the “Average Stats for a Single Season” chart. Finally, we allow users to see which of the two players are more accurate shooters in the percentage charts. Hovering provides more details in each visualization, which are described in the question mark tooltips attached to the bottom of each visualization. By going from the overview charts to the more stat-focused charts, we allow users to make their own judgements on players based on the stats the users value more.

Related Work:

We were originally inspired by other basketball visualization sites, though we were also inspired by a visualization in class. The visualizations presented in the following presentation helped to inspire the parallel coordinates design, in order to show all the seasonal stats in one graph: (https://docs.google.com/presentation/d/1GfEsBYsLoVAX2dpTkxzNP-A3q165gsvR-jIlrASWvkE/edit?usp=sharing). However, we were inspired primarily by these sites (next page):

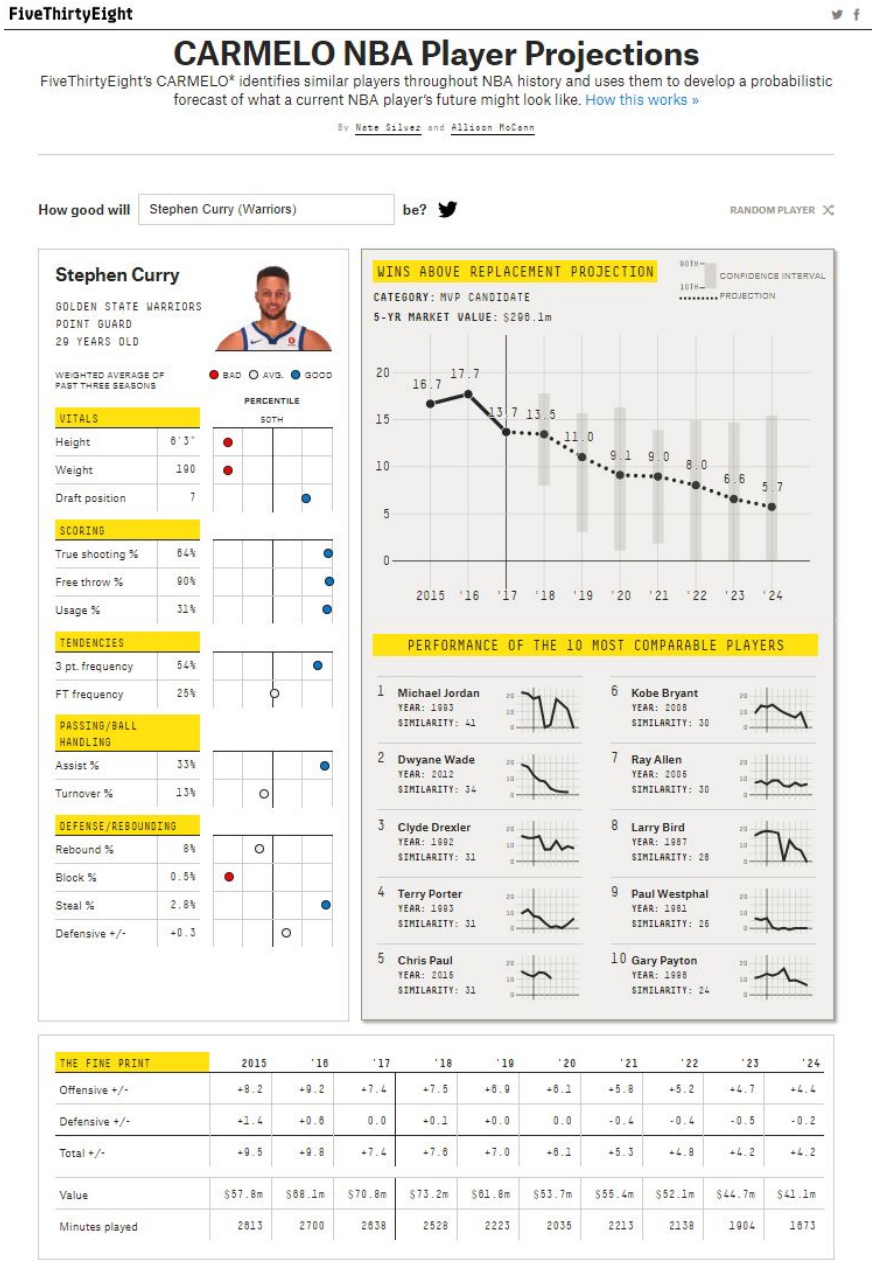
Buckets.peterbeshai.com:



http://buckets.peterbeshai.com/app/#/compareView/201142_2015,2546_2015,2544_2015,201567_2015,201935_2015

This site inspired us to to include some sort of shooting-stats visualization, as accuracy and points scored are two of the most valued statistics. We chose to visualize each player's field-goal percentage, free-throw percentage, and three-pointer percentage as our "shooting-stats" visualization.

fivethirtyeight.com:



THE FINE PRINT	2015	'16	'17	'18	'19	'20	'21	'22	'23	'24
Offensive +/-	+8.2	+9.2	+7.4	+7.5	+8.9	+8.1	+5.8	+5.2	+4.7	+4.4
Defensive +/-	+1.4	+0.6	0.0	+0.1	+0.0	0.0	-0.4	-0.4	-0.5	-0.2
Total +/-	+9.5	+9.8	+7.4	+7.6	+7.0	+8.1	+5.3	+4.8	+4.2	+4.2
Value	$57.8m	$68.1m	$70.8m	$73.2m	$61.8m	$53.7m	$55.4m	$52.1m	$44.7m	$41.1m
Minutes played	2813	2700	2638	2528	2223	2035	2213	2138	1904	1673

https://projects.fivethirtyeight.com/carmelo/stephen-curry/

This site shows the player projections over time, which helped us to define how we wanted to show player progression using line charts to compare the seasons. This especially helped us shape our “Playoff Results” chart, as we compare a player’s playoff results over the years.

Questions:

The primary question we were trying to answer is “Who is basketball’s greatest player of all time?” We compare players side-by-side using four main visualizations; each visualization shows a different aspect of both players’ games. We built this information into basic graphs that the public could easily read while still being informative enough that there would be no room for argument. By building this comparator, we provide the tools that would facilitate the end to these types of debates, based purely on quantifiable stats.

Though we were originally trying to answer “who is the greatest of all time,” we also wanted to find out how time affects players. We wanted to see if certain players, such as Michael Jordan or LeBron James, were able to keep their overall productiveness every season, and if there was some sort of age cutoff where they start to degrade in skill. However, as we progressed, we wanted to know season by season how the players compared, rather than comparing them by age, as the ages would not necessarily mean they have the same amount of experience in the league. For example, some players at age 21 had already been in the league since right after High School (about age 19), so they have two more years of experience compared to players joining the league after some years in college. This experience is hard to quantify but should not be ignored, so we compared the playoff results by each season the player was in the league and included the player’s age as a hover feature.

Data:

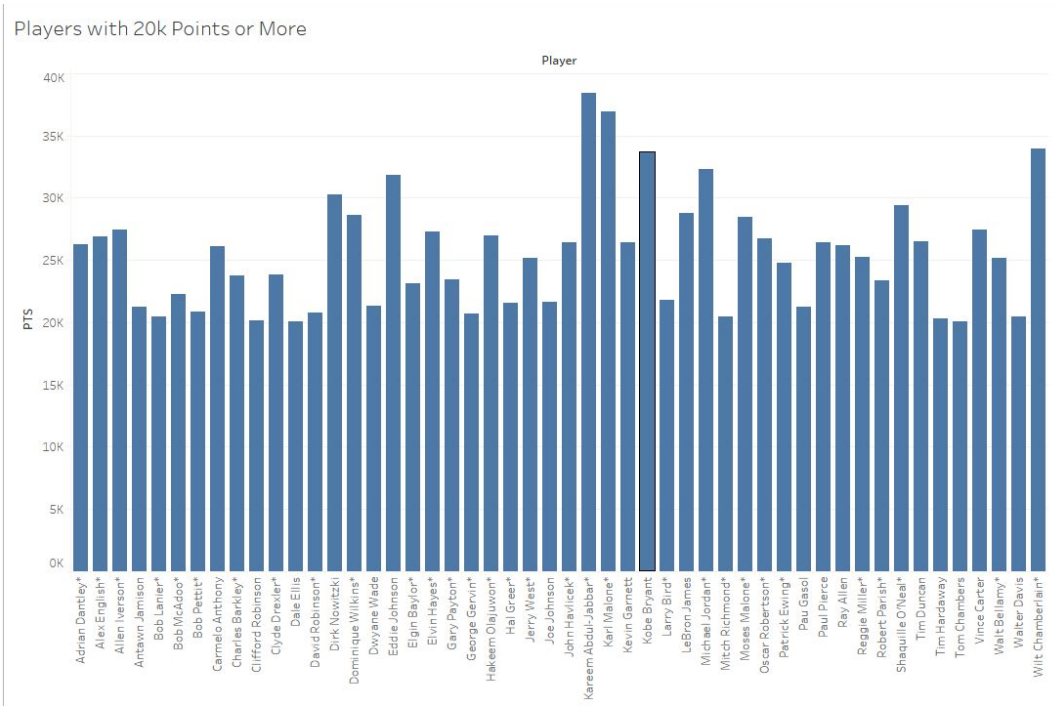
The data was pulled from the Kaggle data set “NBA Players since 1950” which can be found here: https://www.kaggle.com/drgilermo/nba-players-stats/data. We chose this data as it is updated at the end of every season, scraping from the site https://www.basketball-reference.com/ that has accurate data equivalent to NBA’s stat recording sheets, so it is up to date and reliable. Unfortunately, we were unable to include this season’s stats (because this current season is not finished yet), but we would rather have the final stats than an incomplete dataset. Furthermore, the dataset’s records are denoted by the player and the season, which is exactly what we wanted as we wanted to compare players by their career stats, and then the stats by season. For how far that player’s team got into the playoffs that season, we used https://www.basketball-reference.com/teams/.

We filtered the players in order to keep our data set manageable and to make our dashboard quickly loadable. We filtered players based on points, specifically selecting players that reached 20,000 points. Then, as a point of reference for non-NBA fans that have still heard of current noteworthy players (who have less than 20,000 points), we included certain players we believe the general public might know, such as Stephen Curry. Furthermore, we ignored certain stats such as “minutes” played and “games started” as stats such as those are arguably irrelevant. We joined the csvs we gathered, building objects representing each player. Each player is represented by a “player object”. In each “player object”, we have the player’s name, an array of “season objects”, and an array of average stats. Each “season object” contains all of the relevant information for each season, such as the year and the total number of points scored for that player during that season. The “averages” array stores the career average stats for each player.

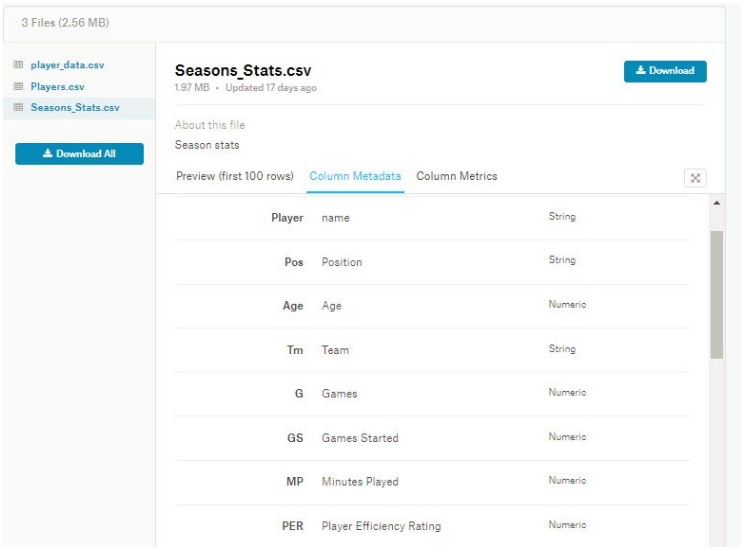
When joining the data in our code, we had to ensure the names matched up. We ran into bugs related to different abbreviations, such as Denver being referenced as “DEV” in one csv while being referenced as “DNV” in another. Other examples include players playing on multiple teams in one season, and missing team names like “San Francisco Warriors,” as some franchises changed home cities and names. Finally, for older players, we had to handle missing values, as certain stats such as the plus/minus score were not implemented and recorded in that player’s time. Ultimately, the joining process and data structure building was done in Javascript and D3, while the initial exploration was done in Excel and Tableau.

Exploratory Data Analysis:

The only exploratory visualization we used was the following bar chart, to compile a list of the players that have reached 20,000 points. We looked at the “seasons.csv” dataset in Tableau, creating our initial list of players we wanted to compare (we later added more players based on outside input):



This was the only exploratory data visualization we used, as the datasets from kaggle included easily understandable metadata that explained the fields/columns of the data:

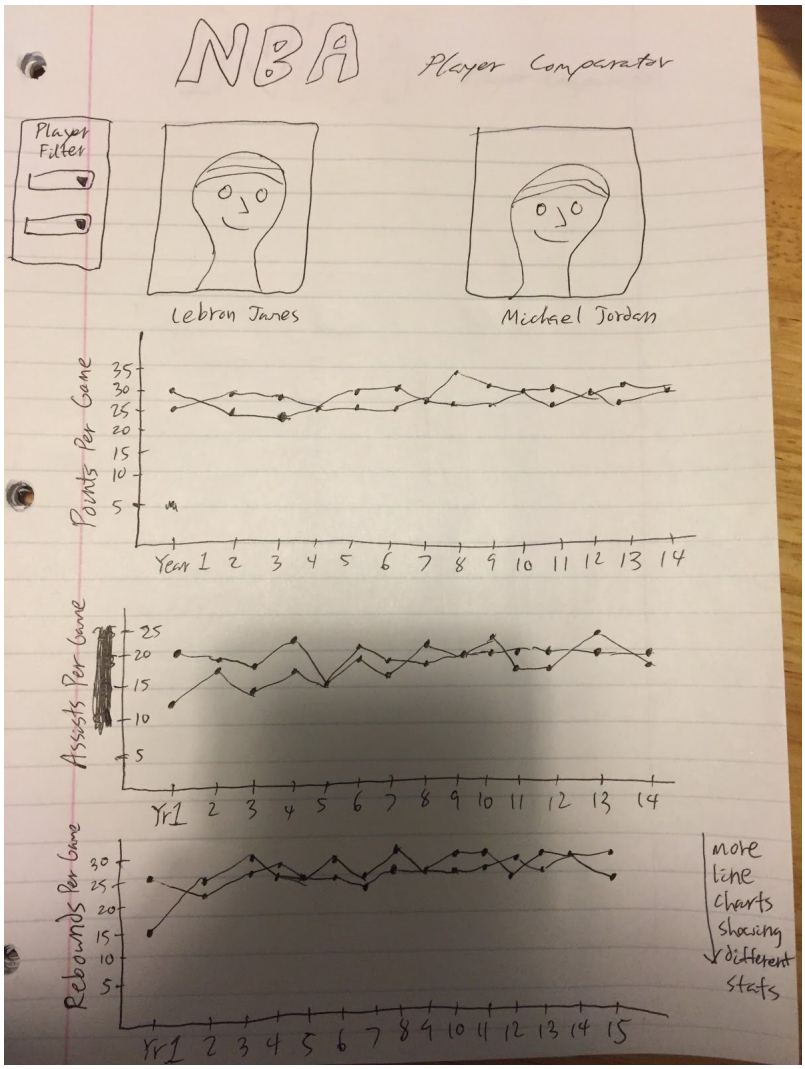


Because we understood how the data was laid out, and because we had a clear goal of which stats we wanted to use, we did not opt to spend a lot of time visualizing the data hoping to find some sort of story. We knew what data we wanted to present, and we luckily found that data in a very clean format, so we did not create multiple exploratory visualizations.

Design Evolution/Implementation:

Though our design did change over time quite significantly, the data we wanted to compare stayed the same, allowing us to keep our internal data structure while our final visualizations changed over time. We considered the following design prototypes, then settled on the final dashboard at the end:

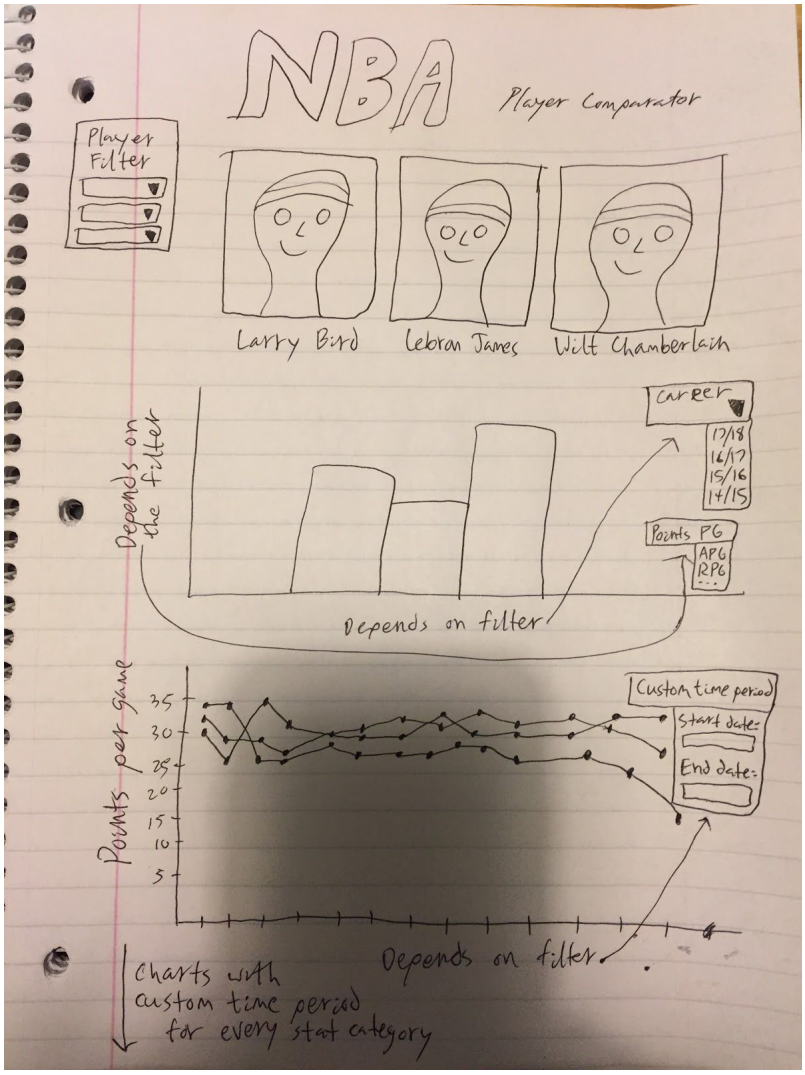
First Prototype:



With this prototype visualization design, users would have the ability to filter any of the NBA players available in the dropdown menus. We were going to show the line charts

for each stat, with the x-axis representing each player's year in the league, while the y-axis represents the stat's relevant scale. Hovering over each point would show the exact value of the stat. This design was rejected because it lacked user interaction and the "detail on demand" principle we were taught in class.

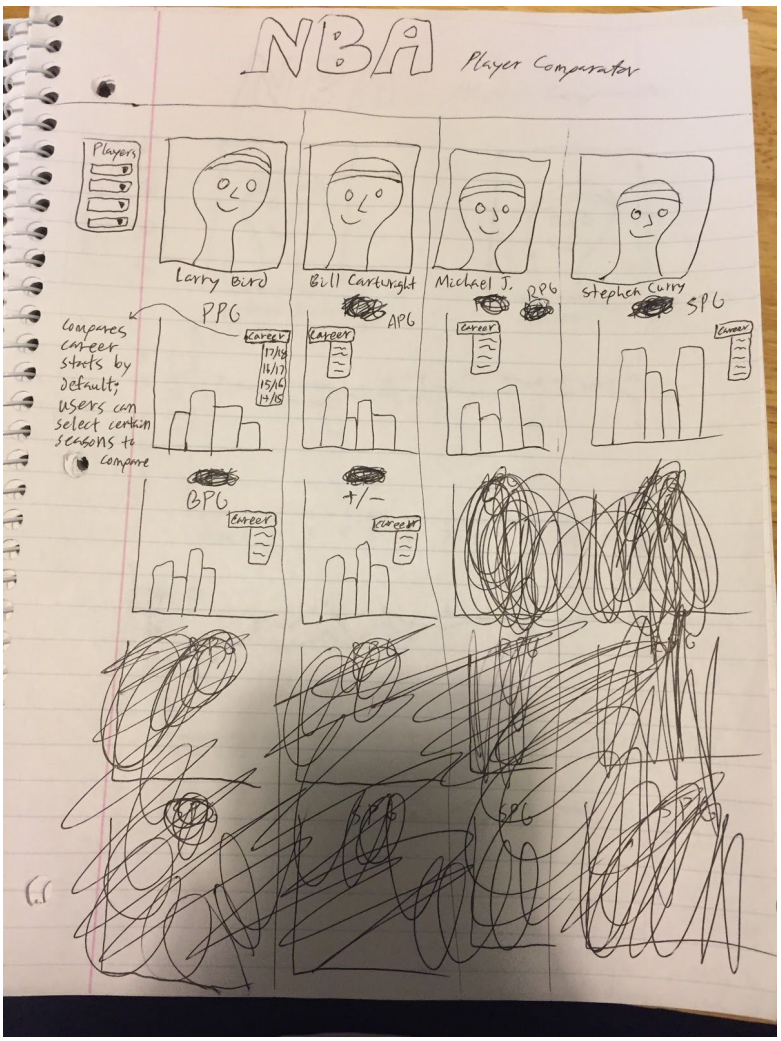
<u>Second Prototype</u> :



With this prototype, users would have the same ability to filter players as the first prototype, but this time we wanted to try comparing three players at once rather than only two. There are two filters for the first "main bar graph": the first filter allows you to choose "Career stats" or you can choose any individual season from the dropdown menu; the second filter allows you to change the stat that is being displayed (points per game, rebounds per game, assists per game, etc….). Users would be able to hover over any bar to get the exact value and to see more relevant information. The rest of

the visualizations below this bar graph would simply be line graphs (one line graph for each stat category), and the user would be able to customize the time period for which they are looking for the average stats. This lacked overall coherence, and the "overview first, then details on demand" principle was not implemented enough to our liking.

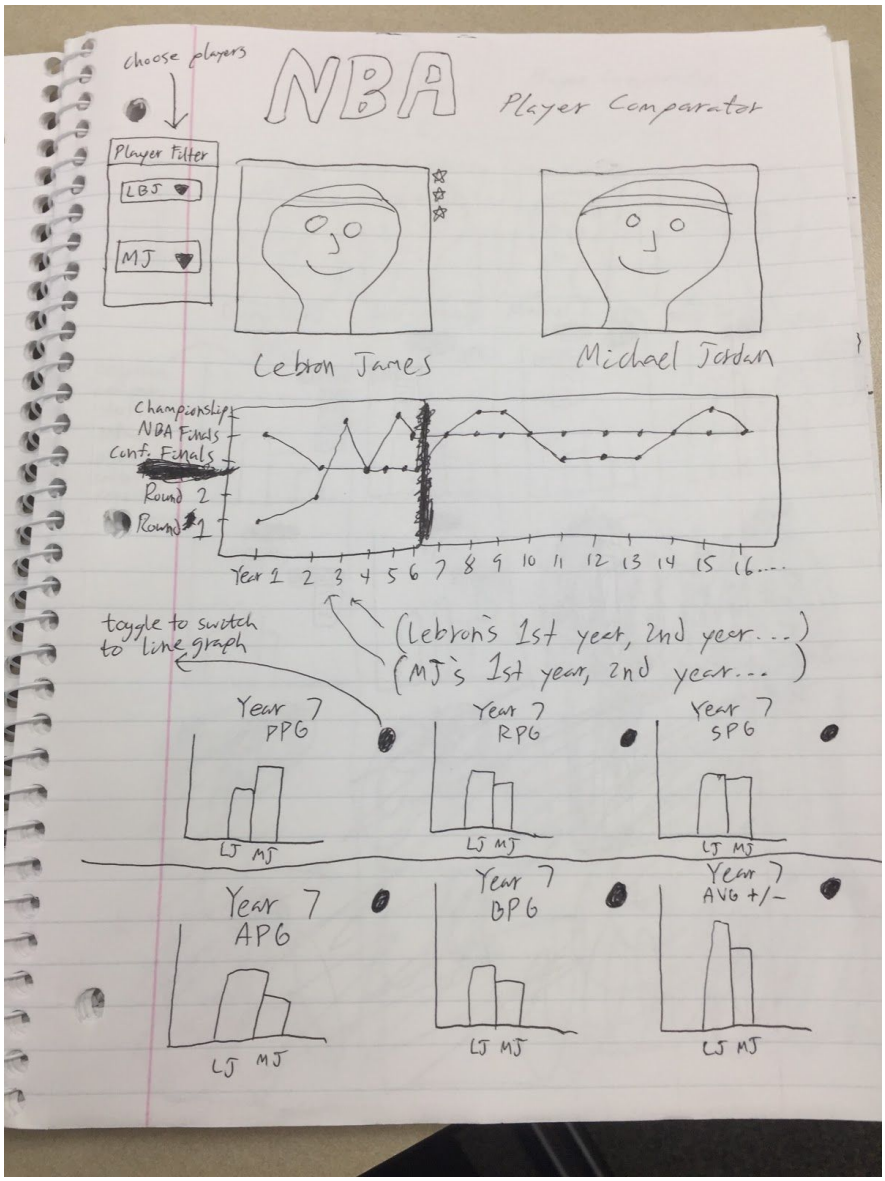
<u>Third Prototype</u> :



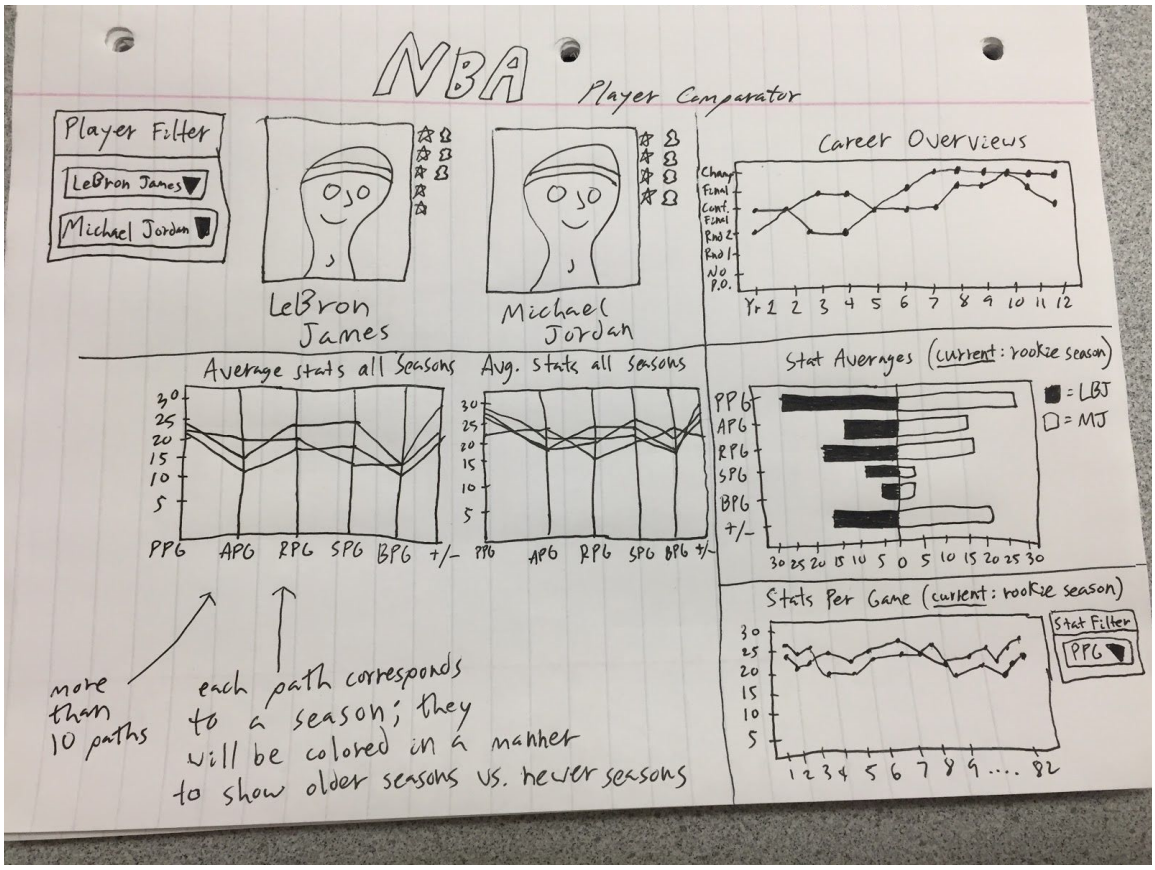
With this prototype, everything that is "scribbled out" would correspond to another bar graph that would show another stat category. We tested out having four players to compare in this prototype. As you can see, every bar graph corresponds to a different stat category. By default, each bar graph would show the average career stats. Using the filter drop-down menu next to each graph, the user would have the ability to select a certain season to compare that specific stat category. This prototype would show a lot of valuable information, but we felt that having four players and try to fit all of these bar

graphs into the screen would make things too “cluttered”. We also felt that we were not using an “overview first, details on demand” approach with this prototype. Lastly, we didn’t think there was enough user interaction or “data exploration” in this prototype design.

Fourth Prototype (original final design, which was changed to fulfill professor requirements on complexity):



Finalized Design:



Here is our finalized visualization design. By default, the two players that will appear are LeBron James and Michael Jordan. Users can choose to compare different NBA players by using the "Player Filter", which features two dropdown menus. Right below the "NBA Player Comparator" title, we show a picture of each player being compared. Below these pictures are two parallel coordinates graphs: each y-axis corresponds to a specific stat, and each path corresponds to a specific season. We plan to color each path (opacity) in a manner that will communicate to the user which paths represent older seasons and which paths represent newer, more recent seasons. To the right of the player pictures (in the next column), we have our "overview" line chart. Here, the y-axis corresponds to how far the players' respective teams went into playoffs (and whether or not the team won the championship), and the x-axis corresponds to each players' first year in the league, second year in the league, and so on. When the user hovers over any of the data points in the overview line chart, the user will see some basic information, such as the year, who they beat and lost to in the playoffs, and the players' respective ages that season. Below our overview line chart is a back-to-back

*bar chart, which by default displays the stat averages for each player's rookie season. This chart would be used to replace the six bar charts we had in our former "final design" prototype, to compare the six basic stats in a compressed manner. Below this is our "Stats Per Game" line chart. By default, it will show the PPG (points per game) for all 82 games of each player's rookie season. The user has the ability to switch the stat by using the "Stat Filter" to the right of the line chart. As an example, by default, the two "sections" below the "overview line chart", which is the "Stat Averages" back-to-back bar chart and the "Stats Per Game" line chart, show information for each player for their *rookie* season. The back-to-back bar chart and the "Stats Per Game" line chart will be updated anytime the user clicks on a data point on the "overview line chart". Also, whenever the user hovers over any text that contains PPG, RPG, APG, SPB, BPG, or +/- anywhere in our design, there will be an educational box displayed that explains the stat category.*

Design Process While Coding:

We built our code incrementally, originally following the final prototype design. We started with the drop-downs for the players, then moved onto dynamically changing the pictures and fleshing out the internal data structure. The parallel coordinates chart was first, then we did the “Playoff Results” chart.

NBA Player Comparator

Player Filter

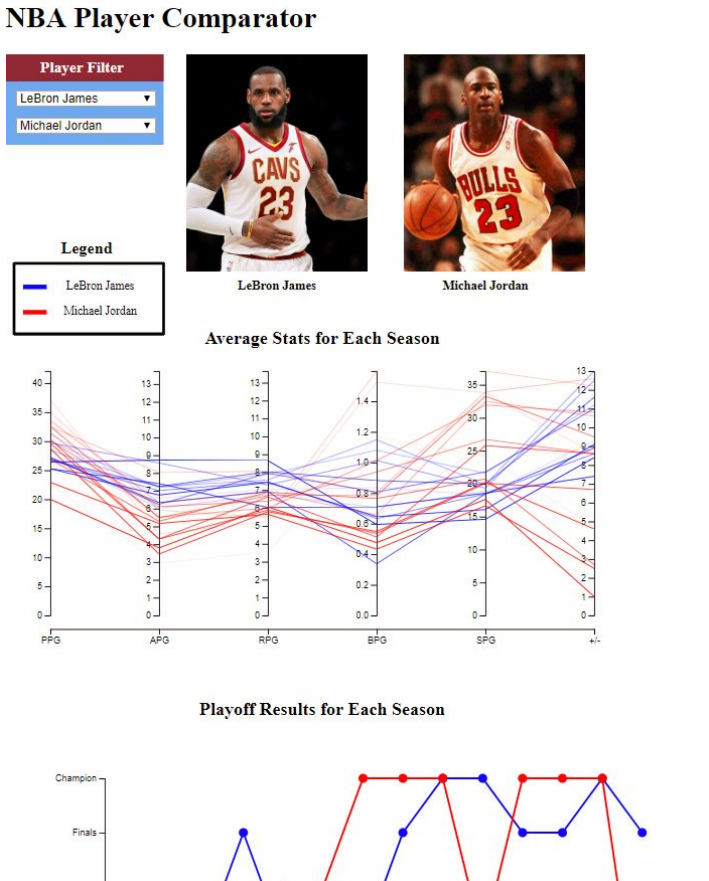
Tim Duncan

Adrian Dantley*

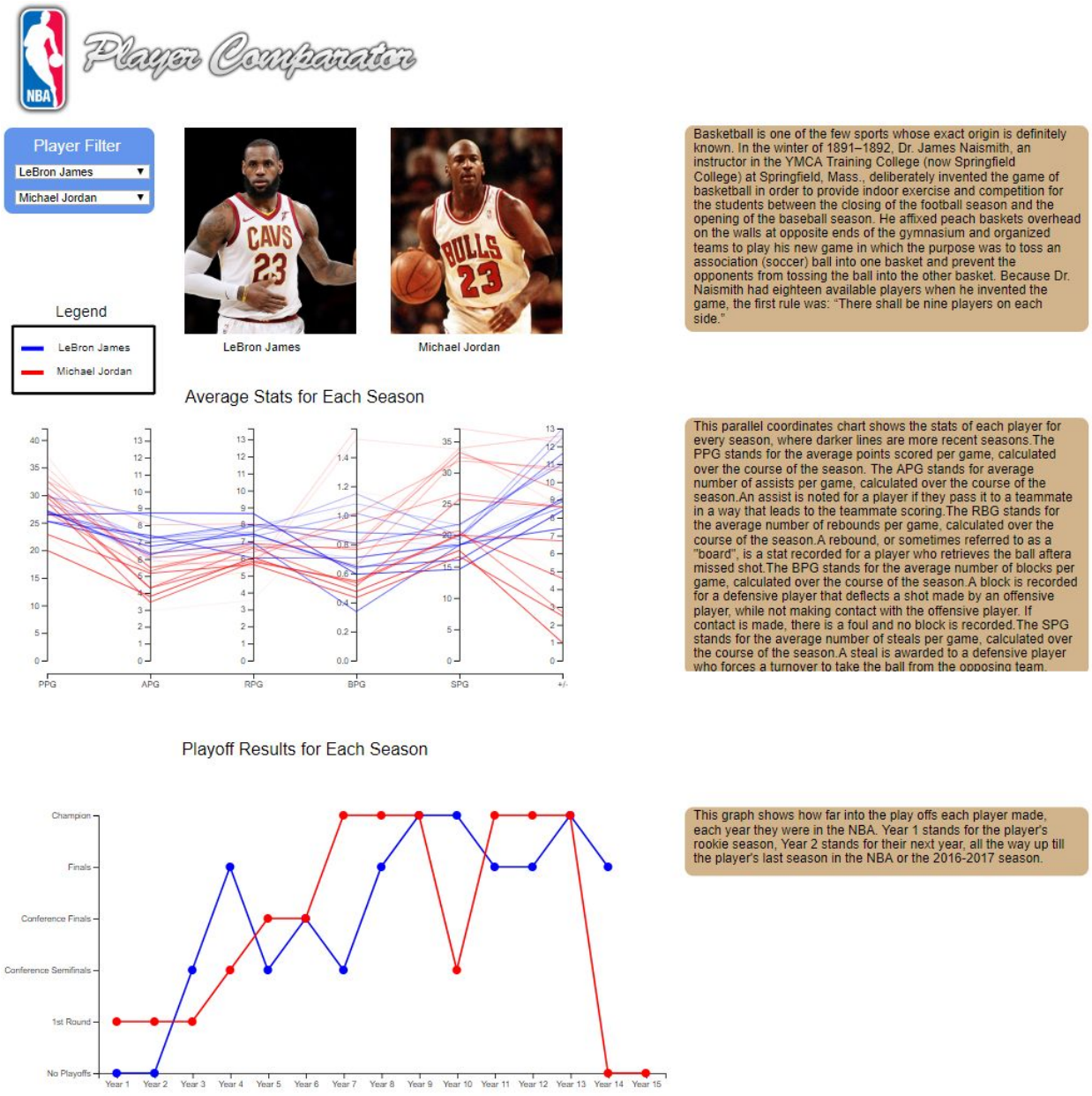
Tim Duncan

Adrian Dantley

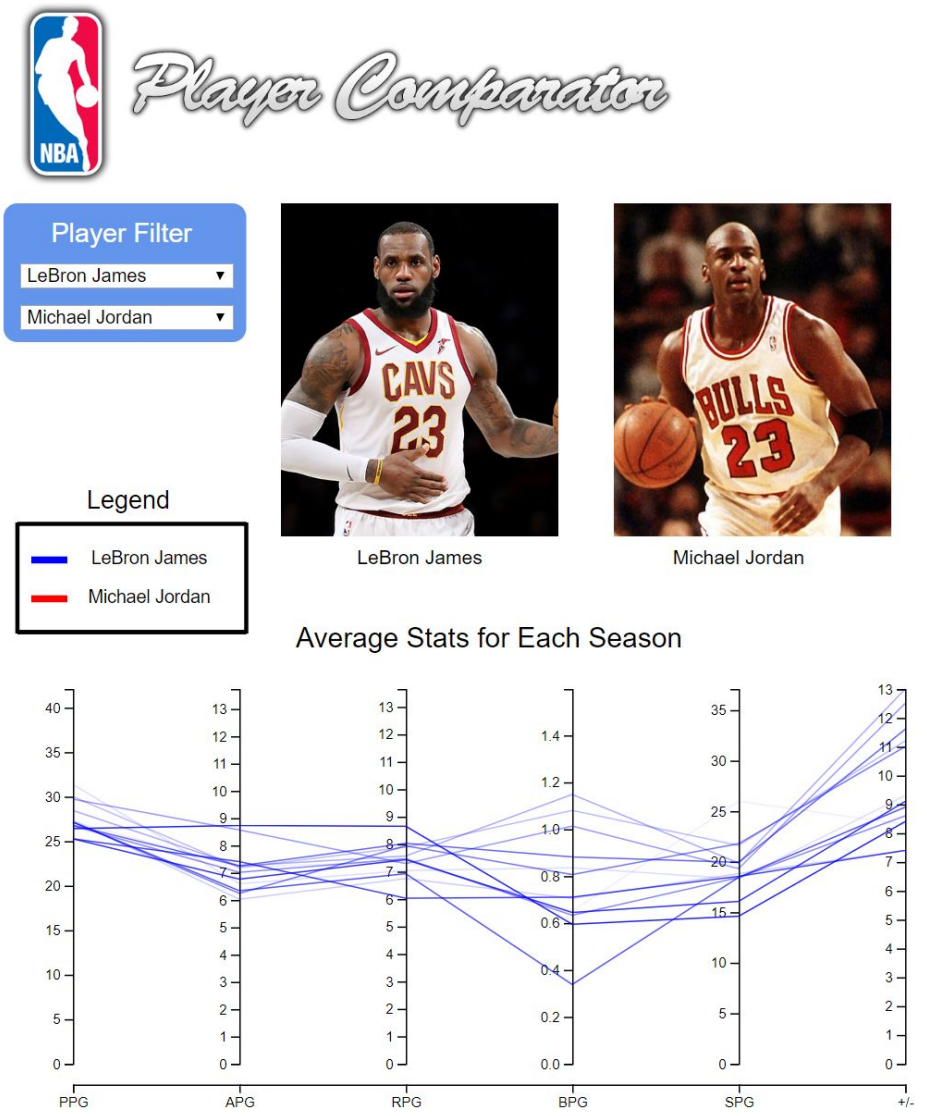
We started with the drop-down filter, allowing us to switch names and pictures based on drop-down changes. After selecting players, users will see the pictures and names change. With these selections, we will also change the following charts dynamically to reflect the stats of the two selected players (start with parallel coordinates chart for “Average Stats for Each Season”, then move onto “Playoff Results for Each Season” line chart, where each circle represents a single season):



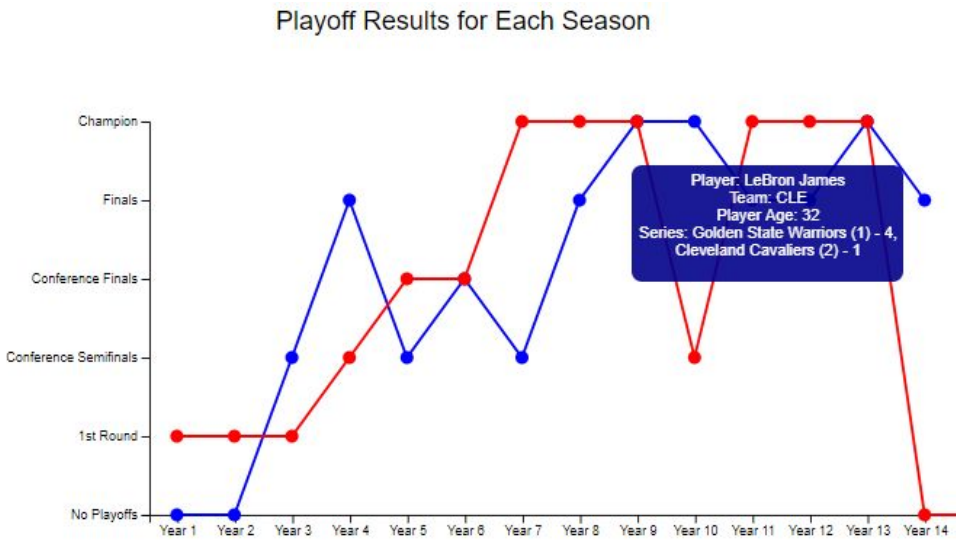
However, we decided to convert to a “scrollytelling” design style as we were recommended by the professor to have more definitions to help the general public that do not know much about basketball. Therefore, instead of having the “playoff” graph to the right of the pictures, we put it directly beneath the “average stats” in order to put definitions and explanations to the right of the graphs:



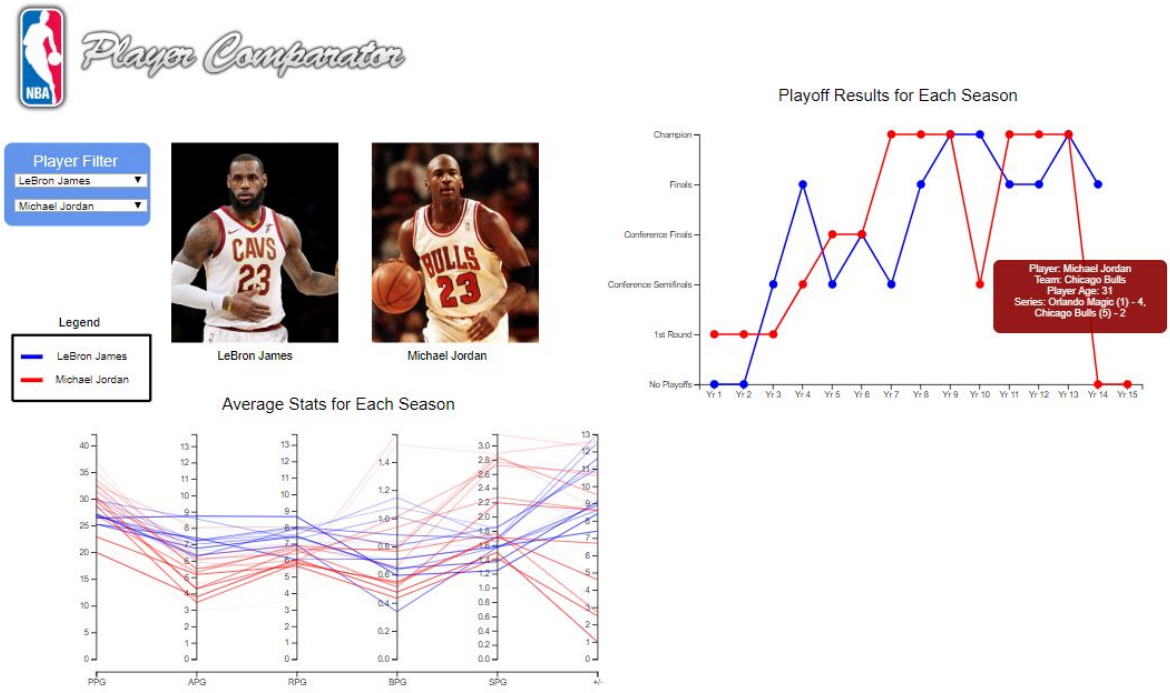
This was our design for the beta release, with some definitions to the right of our graphs in order to provide more information on the features of each graph as well as definitions for basketball terms. At this point in time, we had hover features available for both graphs:



This hover feature involves hovering over the name of the player in the legend, which removes the other player's stat lines in order to isolate a single player's stats. This reduces clutter and allows a user to focus primarily on one player, to look at a particular season or a particular stat for which that player is exceptionally good or notably poor.



The hover feature for the “playoff” graph shows more details on the playoff results, though we were unable to find/scrape data on that season’s record. We instead diverted our time to building out the visualizations and changing the sizes of the graphs to be able to fit everything one dashboard, without descriptions and without having to scroll:



We reverted back to the original dashboard design after feedback from the beta release presentation, leading to the layout above. However, in order to fit any other worthwhile visualizations, we had to shrink the pictures, logo, and graphs, minorly reducing readability. We also got rid of the text boxes, to be later implemented in some sort of smaller tooltip icons that can be be hovered over in order to present more information.

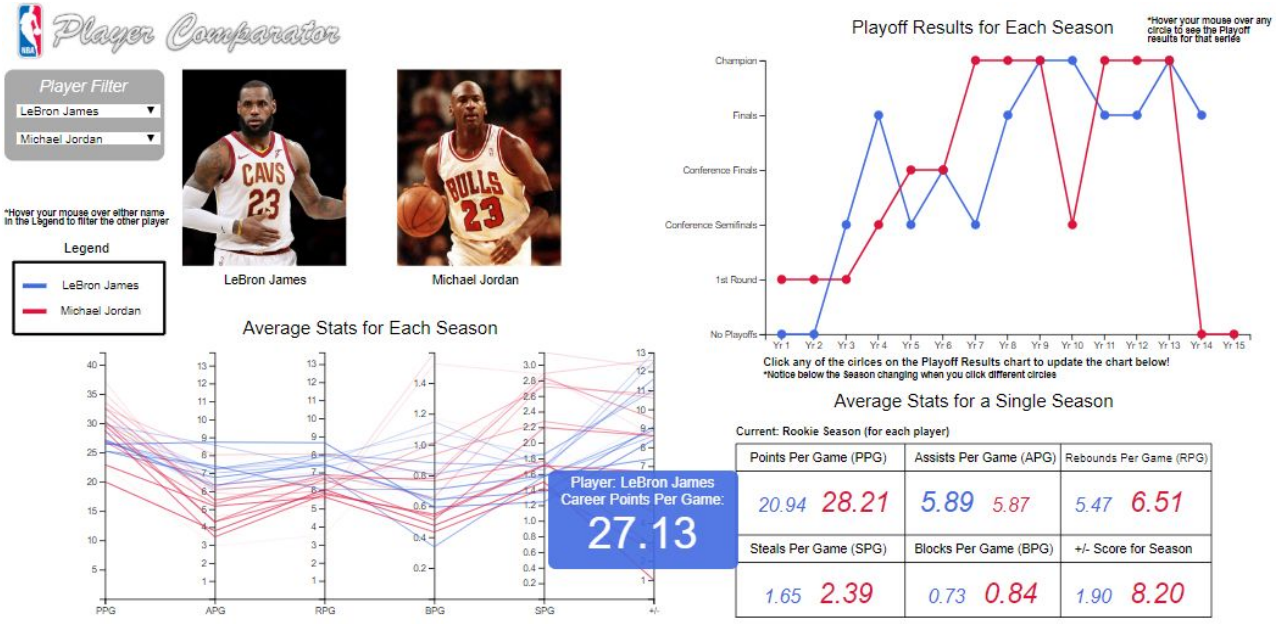
It was around this time that we realized the scales for the back-to-back bar chart would be inconsistent if we tried to fit all the stats in one graph. For example, the “blocks per game” bar would be extremely small, between 0 and maybe 3, while the “points per game” bar would be larger at 20 or more points. This discrepancy would have led to bars that would be hard to read, so we omitted the back-to-back bar chart and chose this table instead:

Average Stats for a Single Season

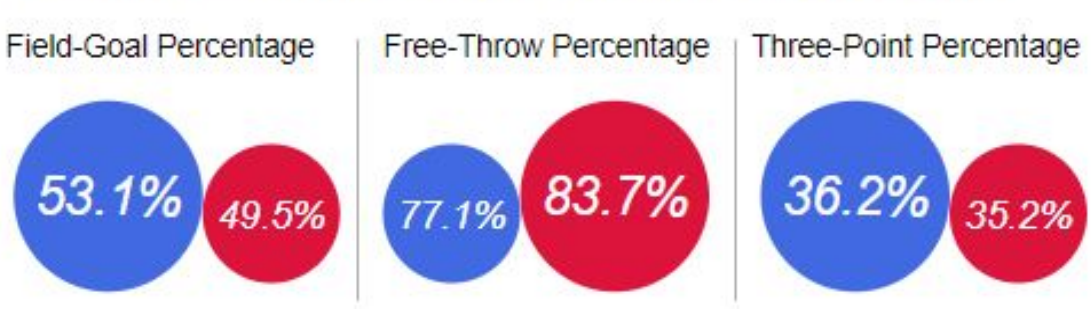
Current: Rookie Season (for each player)

Points Per Game (PPG)	Assists Per Game (APG)	Rebounds Per Game (RPG)
20.94 28.21	5.89 5.87	5.47 6.51
Steals Per Game (SPG)	Blocks Per Game (BPG)	+/- Score for Season
1.65 2.39	0.73 0.84	1.90 8.20

This chart shows the current season selected (defaulted to the rookie season) and shows each player’s respective stat that season. The text size for the stats are hard coded to two single values, with one size reserved for the player with the higher stat and the other size used for the other player’s stat. This was purposely designed because we want to emphasize which player had the larger stat, not the relative difference between the players. This chart updates when a circle is selected in the “playoff” chart.

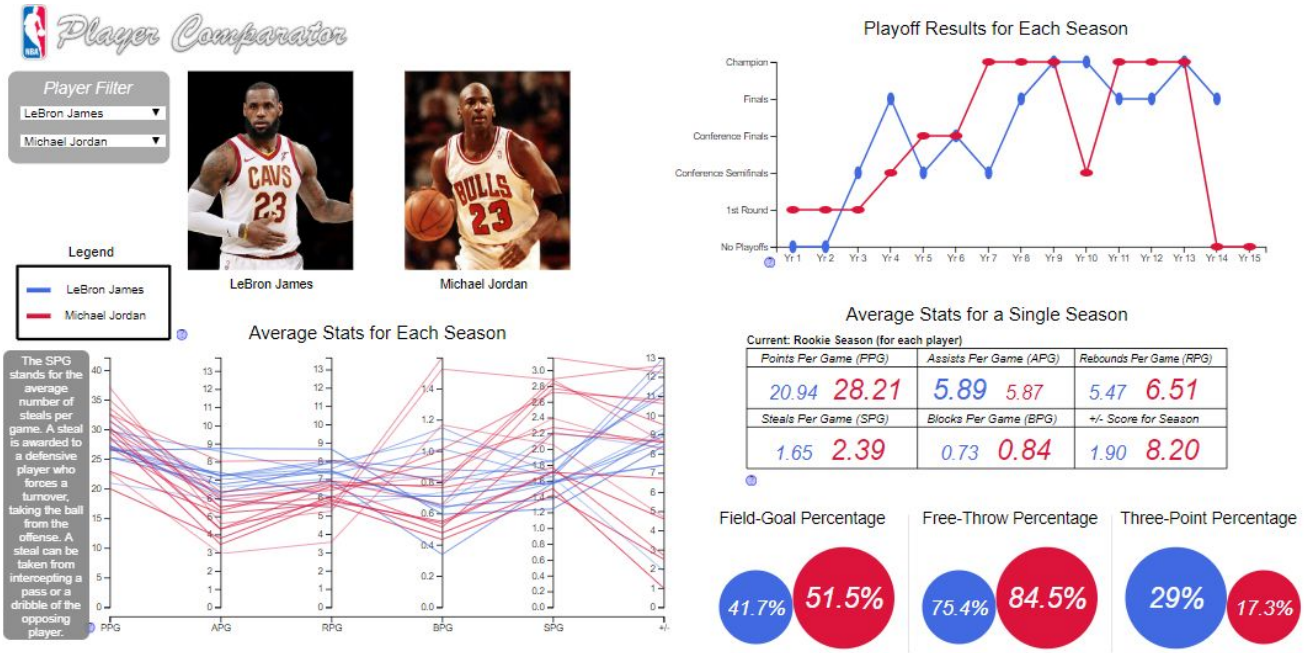


On hovering over each stat, we can see that player's career version of that stat, instead of just the single season's stat. Furthermore, instructions on how to interact with each graph was put into the dashboard, though it looks cluttered this way and led to our final design of question mark tool tips. However, before implementing that, we included one more visualization to round out our dashboard. Because we realized we did not have the game-by-game stats of some of the older players, we decided to change our final visualization from a game-by-game line chart of each stat to a comparison of the shot percentages of the players:

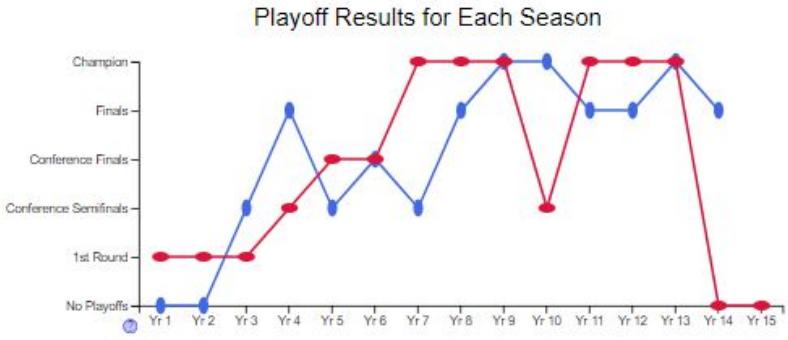


This allows us to break down the "points per game" stat even further, following the "overview first, details on demand" principle even further in our visualization. The shot percentages allows users to see how effective players are, and from where, which allows them to better compare who the best shooters, and therefore best players, are. This also gets updated when a circle is clicked on in the overview playoff chart.

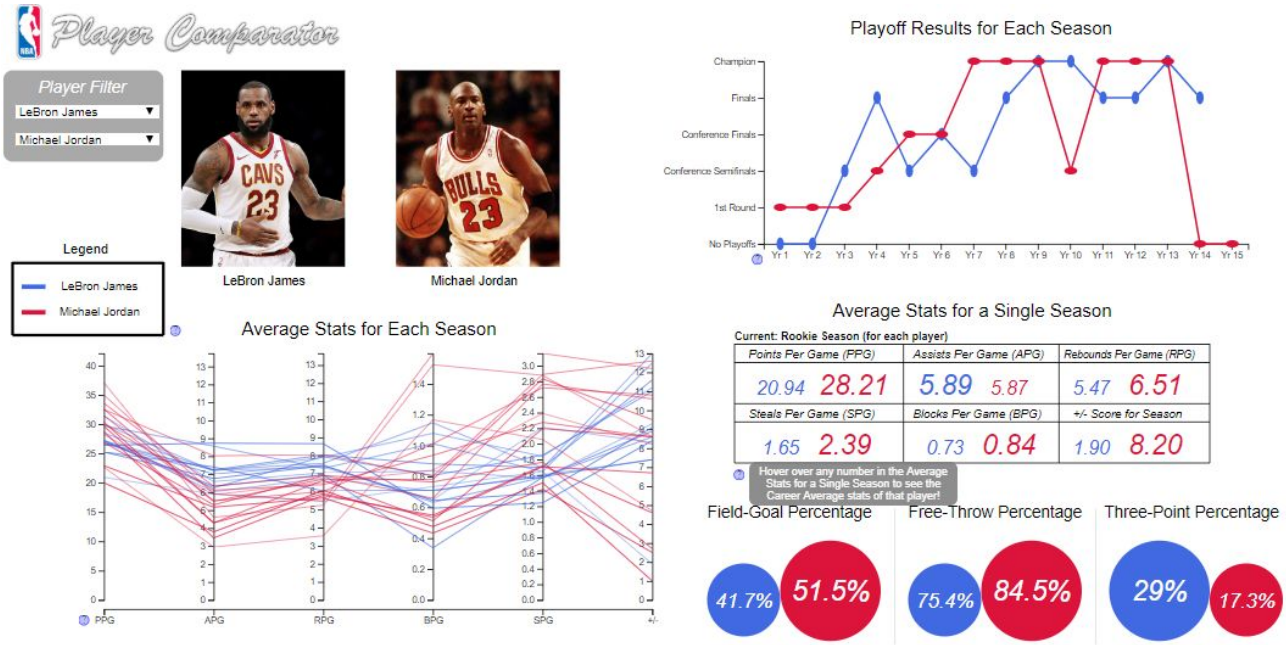
Finally, we implemented hover features for the x-axis for the “Average Stats” parallel coordinates chart to provide definitions for the stats, to fulfill our design of informing new NBA fans and the general public about the meanings of some terms and stats. Furthermore, we replaced the text boxes that instructed users on how to interact with the graphs with question-in-circle tooltips that when hovered on, show instructional text on how to interact with the graphs.



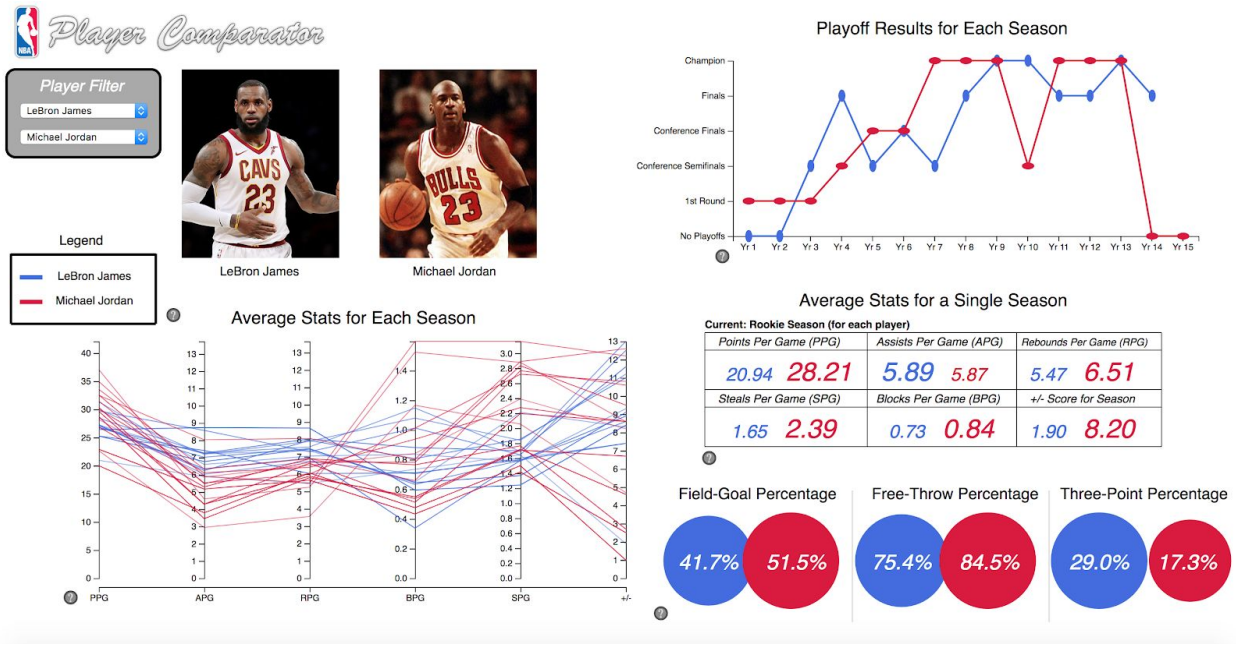
This text box shows up when the user hovers over the “SPG” in the “Average Stats for Each Season” chart. The definitions for each stat will always show up in the white space to the left of that chart.



As a side note, we changed the circles to ellipses in the “Playoff Results” so that coinciding points are still showing, allowing users to hover over the points behind.



Each blue circle with a question mark represents a tooltip. When a user hovers over it, a textbox appears with instruction on how to interact with a relative feature of a graph.



Here is our final design. Initially, the “percentage circles” were given a arbitrary radius value depending on which player had the better percentage. However, we the Professor instructed us to instead use a scale to calculate the radius’ for each “percentage circle”. So, this feature was implemented to set the size the circles based on the difference between the two percentages. We are pleased with the layout and functionality of our final design.

Evaluation:

From our visualizations, we learned that while Michael Jordan had more scoring potential than Lebron James, James did have a higher impact on his team as shown in the higher +/- score. Despite this, Jordan still has more championships, which could be argued as the ultimate gauge of who is the best player. We can see that LeBron seems to be more consistent, as shown by less deviation in the “Average Stats for Each Season” graph, though his career is not over yet, so we cannot say for certain whether he will stay as consistent as he has been the past 14 seasons.

We were only able to set up a framework for answering our questions, as we would still need the entirety of each player’s career to accurately compare two players. We should also take into account other factors, such as injuries, teammates, and opposing teams players, all of which we would need more data from external sources to build a more well rounded dashboard. Other stats we would visualize would be the awards, such as MVP awards or Olympic Gold Medals, won by each player. These would be visualized in icons to the right of each picture, with more space given between the two players to accommodate the additions. In order to keep a single, non-scrolling dashboard, we would include more interactive options to switch graphs to other visualizations, such as possibly a radio button to change the “Average Stats for a Single Season” table to a horizontal double bar chart showing a single one of the stats.