DATA 511: Data Visualization for Data Scientists
December 13th, 2022
Team Members: Peter Mohammadi, Ramiro Steinmann Petrasso, Rthvik Raviprakash, Vaibhav Mehrotra, Yamina Katariya, and Yashwanth Manne

C4: Concept Production

Dashboard: Tracking NBA Payroll and Performance (1985-2022)

1. Executive Summary

In the National Basketball Association (NBA), teams are only allowed to allocate a certain amount of money for their total payroll. Oftentimes better players get higher salaries while the younger/worse players get paid a lower salary. NBA fans are not oblivious to this practice, but in actuality: are the best players always getting paid the most? Are the most successful teams spending the most money? And historically, have the teams that won the championship even spent their money efficiently? We created a dashboard in Tableau that explores these questions. Our dashboard lets users select any team in the NBA from 1985 to 2022, dive into the team's performance metrics, and evaluate any box score data and shot map for any team member in the chosen season. By exploring this dashboard, viewers can quickly notice that, in many cases, the best players are not paid the most on the team. Thus, there is not as strong a correlation between salary and performance as we, and other NBA fans, expected. We sourced data from the official NBA website, Basketball Reference, a reputable source for NBA player data, and FiveThirtyEight, a popular data analysis company.

2. Concept Background

Our visualization aims to explore the link between an NBA player's salary and their overall performance and find patterns between the two. The NBA is one of the world's leading men's basketball leagues and generates billions of dollars across the organization annually. Analyzing the performance of teams and players relative to their income can help NBA fans better understand the organization and players they follow. To evaluate the performance of teams and players and their salaries, we gathered and explored RAPTOR, Elo, salary, shot efficiency, and per-game player statistics data across 38 NBA seasons. For our analysis, we focused only on regular season data to allow for even comparison among teams; all teams played 82 regular season games, but not all teams played in the playoffs.

2.1. Data Acquisition

After exploring the different NBA player performance metrics available (via both official and unofficial sources), we found one interesting measure that could help us compare players holistically. That measure was the RAPTOR score, a metric put forward by Nate Silver from FiveThirtyEight. RAPTOR, which stands for **R**obust **A**lgorithm (using) **P**layer **T**racking (and) **O**n/Off **R**atings, was designed to evaluate NBA players similar to how modern team managers would evaluate their players using only publicly-available data. The metric primarily combines box score data – a summary of points, rebounds, assists, steals, blocks, and shot efficiency – with on/off metrics, which tracks how much better the team performs when the player is with him than without him. Additionally, RAPTOR also considers categorical data such as height, draft position, and whether or not the player has competed in an All-Star game. Silver notes that a team RAPTOR score can be obtained by simply summing up the scores of a team's players. Thankfully, RAPTOR data is continually maintained by FiveThirtyEight and is available to download as a single CSV file.

In addition to RAPTOR, we used the Elo ratings of teams to represent their success. Essentially, Elo compares the relative skill levels of two teams in zero-sum games. If a team wins a game, their Elo rating increases based on the relative skill level of the opponent; wins against weaker teams result in smaller increases than wins against stronger teams. Since Elo ratings are zero-sum, the loser will always lose the same points the winning team receives. Thus, Elo exclusively tracks wins and losses & opposing team strength. As with the RAPTOR data, Elo data was present as a single CSV file on the FiveThirtyEight website.

After extensive research, salary data proved difficult to find. The only available source was Basketball Reference, which contained historical salary data as individual player profile pages tables. So, we needed to scrape tables off of player pages and aggregate them into one large data set. Additionally, we had to consider a way to standardize the salary data to account for inflation and league revenue. The easiest way to do this was by downloading a readily available CSV of league salary caps over the past 38 seasons & scaling each player's salary by the league salary cap at the time.

Per-game seasonal averages for each player were also sourced from Basketball Reference, which publishes individual tables of player stats for each season. We had to grab each season's table, download it as a CSV, and merge them with a column identifying the season.

The easiest way to collect shot data for each player was through `nba_api`, a Python package that creates API endpoints that query information from https://nba.com. Specifically, we used the `shortchartdetail()` function from that package using five parameters: player, team, season, season type, and context (category of shots). For our analysis, we tested all player-team-season combinations in the RAPTOR data for regular season games focusing on the locations of shots made and attempted.

As we began building the dashboard, we realized we wished to include team logos and player images to help the audience feel more connected to the dashboard. Thankfully, Basketball Reference includes an image of each player on their respective profile page, and we were able to scrape them easily. The team logos were manually downloaded from the NBA website.

2.2. Data Handling

All data files were stored locally as CSV files since the total data size did not exceed 1 GB. The largest data source was the NBA shot data, which needed to be stored as a compressed RAR file. Since most of the data sets were small, they were stored on GitHub, allowing all team members to pull the latest files for subsequent analysis. To avoid tampering with original data, cleaning scripts generated modified data files that standardized conventions for 3-letter team abbreviations, player IDs, and season. These three primary keys allowed for the merging of all collected data files. All data cleaning and subsequent analyses were done in Python using Pandas, Numpy, and regular expressions.

2.3. Data Preparation, Problems, and Modifications

We used the official NBA acronym as the three-letter abbreviation for each team. Since the data sets originated from multiple sources, we mapped the full-length team name to our desired convention. Additionally, since multiple older teams were later bought out/relocated to the existing 30 teams, these mappings also needed to be done manually in each data set.

Since the salary cap data, essential for the salary analysis of our data, only existed from the 1984-85 season onward, all additional data sets were filtered for the time range of the 1985-2022 seasons. Additionally, because the "1984-85" season is conventionally referred to as the "1985" season, all season columns across all datasets were converted to the latter year. This also changed the variable from an ordinal string variable to an interval numeric variable, making it easier for downstream analysis.

Player IDs were not consistent among all data sources. Since most of the data files were from Basketball Reference or–in the case of RAPTOR & Elo ratings–developed from Basketball Reference data, we encoded the Basketball Reference player ID as the main ID key.

The salary data of each of the players was extremely difficult to scrape. There were no ready-made APIs to call, and we relied on BeautifulSoup4, a Python library, to scrape information from web pages. Additionally, the salary tables were views of backend tables, whose links were no longer publicly available. Thankfully, one of our team members had previously scraped the data & he could still access these tables with a deprecated link format to the salary sources. There was still a considerable amount of missing values in the salary data, but a quick perusal showed that these were for players with negligible amounts of playing time. Once salary data was merged, the salaries of each player were scaled by the league salary cap of that season. These scaled salaries followed a log-normal distribution and were transformed into a normal distribution for downstream analysis.

The CSV of Elo ratings was structured as individual pre-game & post-game scores for every game ever played. Thus, the data needed to be parsed with Pandas in Python to generate an aggregate Elo metric of each team for each season. Specifically, we took the average post-game Elo rating across all 82 regular season games to generate a representative Elo rating of a team for each season. Since each season's game logs go through the end of the NBA Finals, we could also scrape the NBA champion for each season based on the winner of the last game.

The RAPTOR data was relatively clean, with no missing values. Still, the RAPTOR metric was susceptible to outliers from players who played a low number of minutes. So, we weighted the RAPTOR of a player by the number of minutes played to generate a newer RAPTOR that conserves all the properties of the initial RAPTOR while removing spurious outliers with small sample sizes. Additionally, RAPTOR was normally distributed by design & the minute-weighted RAPTOR remained normal.

It took a lot of work to figure out how to present a primary comparison of team performance and salary, especially with missing salary data. After much deliberation, we realized that since RAPTOR & salaries can be distributed normally, we can compare the two with Z-scores. So, we compared an individual's performance with and salary by subtracting the Z-score of the normalized salary from that of RAPTOR. Then, we summed the Z-score difference for all players on the roster to generate the overall team metric for performance w.r.t. salary.

Shot data was stored as separate JSON files for each player-team-combo. As such, it took considerable time to parse and merge the data into one cumulative CSV file. Additionally, the API calls were especially susceptible to Unicode errors with accents on names, typos, and punctuation (James Jr vs. James Jr.) in the player name, generating a slew of erroneous calls. Some players also had slight mismatches in their names (Stephen vs. Steve). So, we first reran the API calls after fixing the Unicode errors with a handy Python library named Unidecode. We manually analyzed the data for subsequent failed calls and found the official NBA player ID for player names with errors. This vastly reduced the amount of missing data. Still, it should be noted that shot data was not tracked before the 1996-97 season.

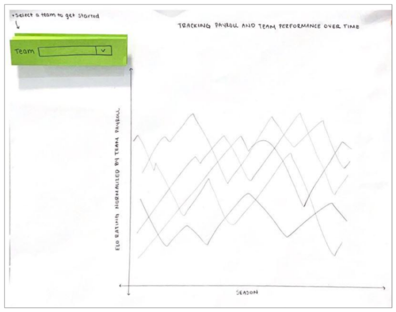
Next, once the cleaned data sets were loaded into Tableau, we still needed a way to visualize points as a heatmap. To do so, we used hex bins to cluster shots taken close to each other to give a better aggregate. We had to try multiple sizes of hex bins to get the most visually aesthetic and easy-to-interpret heatmap. The most challenging part during this step was understanding the unlabeled units of the X & Y dimensions of the shot data and mapping the data to a picture of an official NBA basketball court.

To place the per-game player statistics in context, we took the weighted average by position of each of the main box-score stats based on the number of games played for each season. A quick search of averages on Statmuse, a known NBA data source unavailable for download, matches our calculated values.

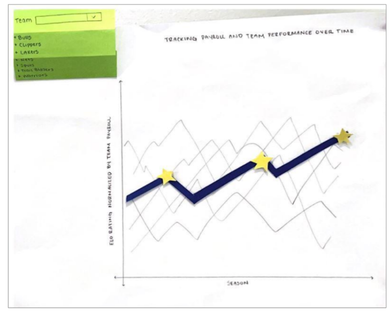
Approximately 1000 player images were not possible to scrape from Basketball Reference. These were manually downloaded from internet searches. All other data sources were error-free without missing values.

3. Design Usability Testing

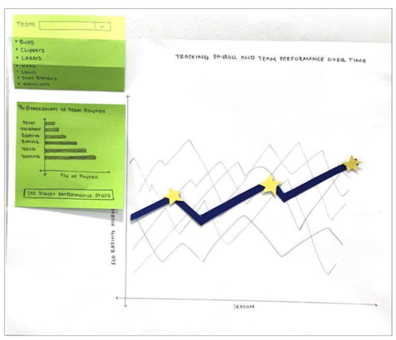
3.1. Initial Testing Results



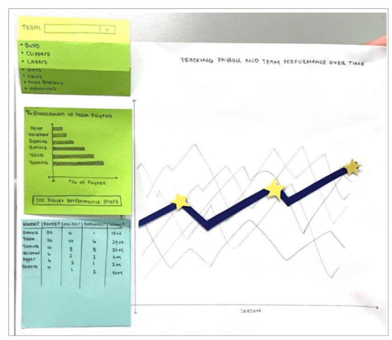
Initial View
All Performance w.r.t Salary curves for all teams are shown.



Task 1
Selecting Golden State Warriors (GSW) in the the team dropdown highlights only the GSW curve with stars representing championship seasons.



Task 2
Clicking on a star shows salary of each of the players in the team as a sorted horizontal bar chart. Additionally, a button is present to see individual player stats.



Task 3
Clicking the individual player stats button results in a sortable table for each player's performance metrics.

Figure (1). Overview of Low-Fidelity Prototype

For our initial prototype, we interviewed three people to test the overall structure of our dashboard. The tasks that we picked for our users to complete were: (1) filter the teams to pick the Golden State Warriors, (2) choose the team's most recent championship season and identify the payroll of a player for the team during that season, and (3) find the top three scorers on the Warriors during that season.

All of our in-class usability testing subjects could complete the first task easily. They all recognized that the team list box was used for filtering and, therefore, could filter for the Warriors easily. The second task was where we saw users get confused. Some could deduce that the stars on the highlighted line for the Warriors were championships but indicated it was not intuitive. It was clear that labels on the bar chart did not clearly represent the percentage of payroll as some thought they were points, and all were confused about what the y-axis indicated.

3.2. Beta Testing Results

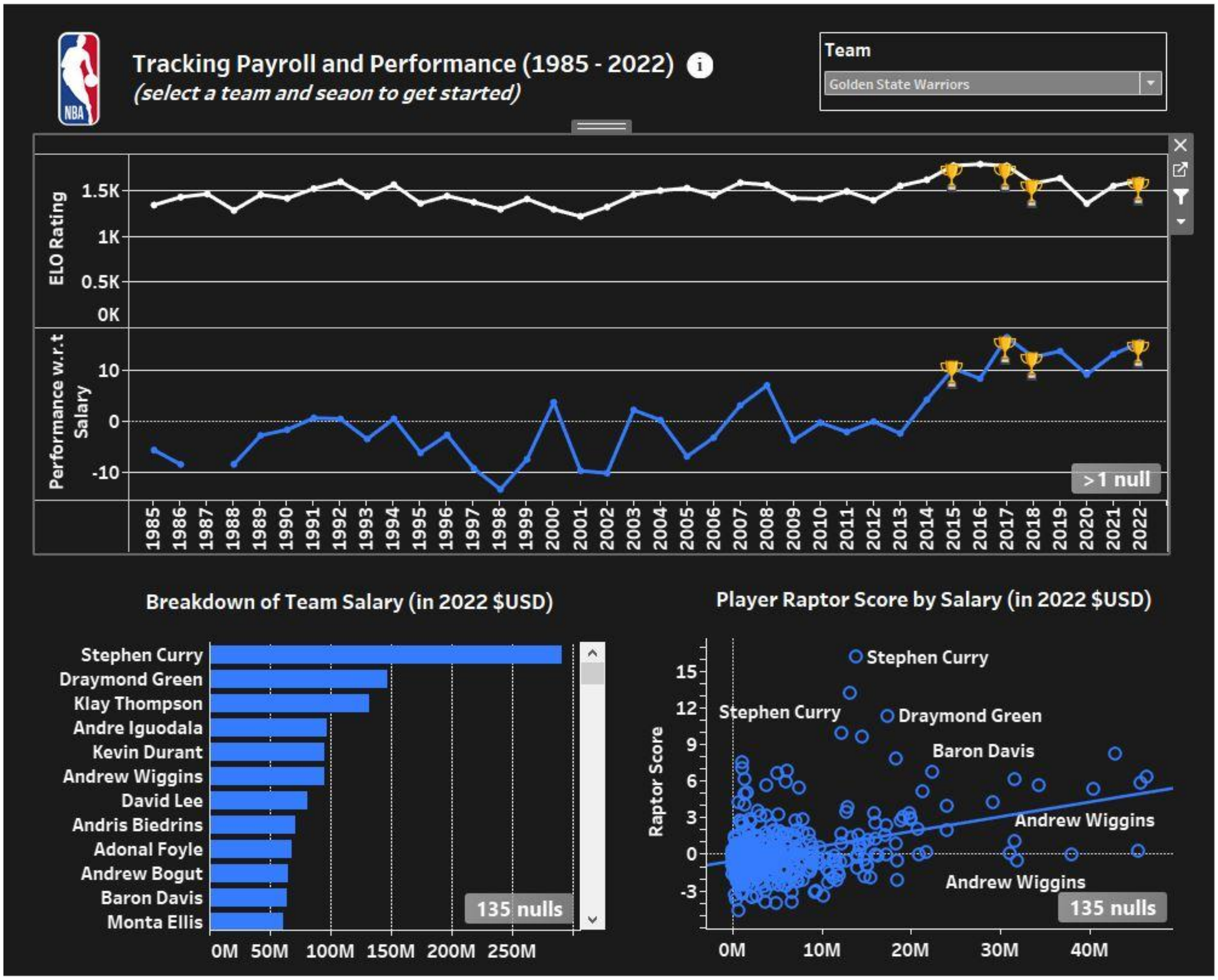


Figure (2)a. Dashboard Home Page - Beta Version

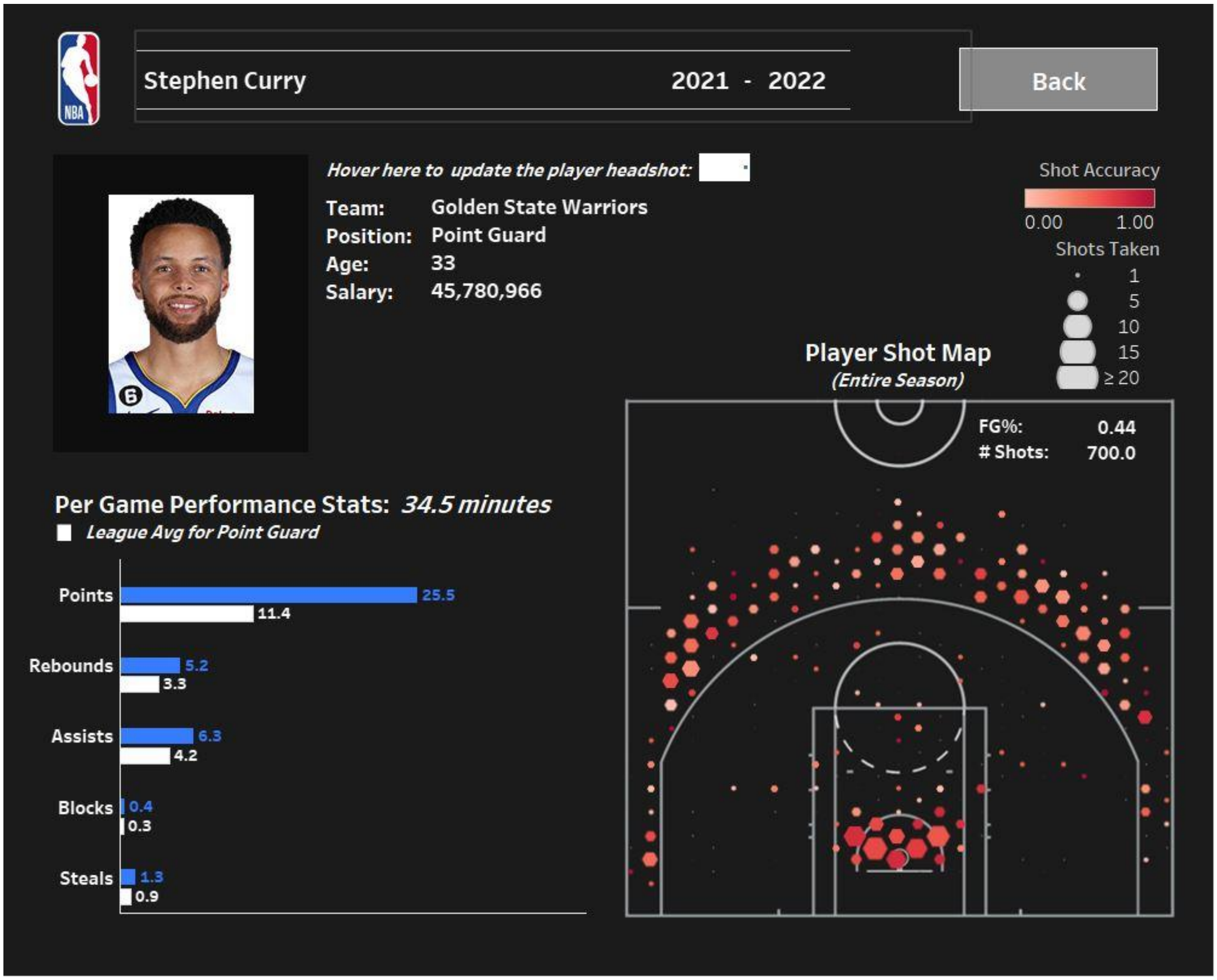


Figure (2)b. Player Statistics Dashboard - Beta Version

Our beta testing involved interviewing six avid basketball fans. Overall, the selected group loved the look of the dashboard–especially how the dashboard's color scheme changed to the selected team's. They also liked the data we were showing them; they all felt like our metrics correctly represented the skill of a player/team. However, there was also common negative feedback; a handful had issues with the dashboard flow, not knowing what or where to click to get more information. Additionally, they suggested some small edits to the legends.

4. **Initial Design Problems and Modifications:**

The original marker for a team's championship win was a star. This caused confusion among initial testers, so the marker was switched to a trophy instead. As most of our usability test subjects were unfamiliar with several of the terms used in our visualization, we included a *Definitions* tooltip. We also included an *Information (i)* tooltip next to our *Player RAPTOR Score vs. Salary* graph that explains the main takeaway the trend line provides.

When displaying each player's statistics, we wanted to include their headshot to help the audience identify the player. Due to unforeseen technical difficulties with external link securities, we were unable to render webpage objects of player headshots when publishing to Tableau Public. The feature worked when we ran the dashboard locally, but when it was deployed, the dashboard was denied access to the pictures due to the CORS rules of the pages hosting them. As such, we pivoted to including only the team logos as custom shapes & a button linked to each player's Basketball Reference profile. We also bypassed the embedded webpage of player images issue by linking to an external player profile webpage on Basketball Reference.

5. Insights and Breakthroughs

It was during our presentation that we discovered some areas of our visualization that needed more clarity. We discovered that when the user selected a season on the main line charts, it was not displayed anywhere on the visualization that they had selected that specific season. In order to mitigate this issue, we included a *Season Selected* legend that indicated to the user which season they selected on the graph based on general feedback.

We were also notified that it would be informative if users could compare multiple teams. This would further let the user compare the salaries and Elo ratings of the specific teams that the user wishes to see. In response, we added another dashboard that allows users to compare multiple teams' Elo ratings and performance w.r.t. salary and zoom into these comparisons by filtering the range of the seasons shown on the graph.

The most significant response given during our presentation was that the *i* tooltip that provided definitions was extremely helpful and appreciated. In response, we added a clearly labeled *i* tooltip and a How To tooltip that gives instructions on interacting with our dashboard. It was clear that not only was the terminology unfamiliar to our audience but the flow was also a bit unclear.

6. A Critical Evaluation of the Project's Success

6.1. Critique of *Elo Ratings & Performance w.r.t. Salary Over Time* Line Graph

The graph shows a trend of how the Elo ratings and performance w.r.t salary have changed over the years. Championship seasons are highlighted using trophies allowing viewers to make effective comparisons that would not have been obtainable without being denoted as such and ensuring the graph is aesthetically pleasing and memorable. The *Elo ratings & Performance w.r.t Salary Over Time* graphs allow users to filter the visualizations by clicking on data points. This action filters our other graphs specifically to data for the selected year. Filtering data values is intrinsic to the visualization process, as analysts rarely visualize the entirety of a data set at once (Heer & Shneiderman, 2012).

The best way to show ratio data, Elo ratings & performance w.r.t. Salary, against interval data, season year, is to use a line graph that encodes the visual attribute of position & slope. The graph successfully highlights important information by allowing users to filter to a specific team and denoting when that team won a championship. The lines are color-coded based on teams and further properly represent data by clearly mentioning the labels and legends for both axes.

6.2. Critique of *Breakdown of Team Salaries* Bar Graph

We used a horizontal bar chart to display the breakdown of each team's salary for a specific season. Here, player salaries and ratio data were plotted against the nominal data of player names. During week 2 of this quarter, Professor Mannheimer noted that horizontal bar charts were slightly more effective than vertical bar charts due to the ease of vertical comparison (Mannheimer, 2022b, slide 92). Plotting this bar chart horizontally and coordinating the color of the bar charts with the team color ensures the audience can efficiently make player salary comparisons while quickly associating the entire graph with the team they are interested in. The salaries were also sorted from largest to smallest to highlight players with the largest salaries and accommodate the audience's natural flow of viewing a graph. This graph links each player to another dashboard. When you click on the bar of a specific player, a menu is displayed, which contains a link that takes us to another set of visualizations that showcase player statistics and their information, following a view manipulation task (Heer & Shneiderman, 2012).

6.3. Critique of *Player RAPTOR Score vs. Salary* Scatterplot

We selected a scatterplot as both values plotted are ratio data, and position is the most effective visual encoding for quantitative data (Mannheimer, 2022b, slide 17). As some points on this graph occasionally overlap, we used hollow circles for the shape so the audience could differentiate between overlapping points.

6.4. Critique of *Per-Game Performance Stats* Bar Graph

This horizontal bar graph compares the selected player's per-game statistics and the average statistics of other players that hold the same position for the selected year. We use the same rationale for a horizontal bar graph as described in section **6.2.** As the user was filtering to look at a specific player, we did not feel it was necessary to explicitly label which color denoted the player as we included a color legend for the average player with the same position. This was done to remove clutter and unnecessary writing (Mannheimer, 2022c, slide 53). To further remove unnecessary text, we included the values of the bar chart at the end and removed column lines. This maximizes the Data-ink ratio (Mannheimer, 2022b, slide 46).

6.5. Critique of *Player Shot Map*

The player shot map, which is a heatmap overlaid on top of a basketball court in the background, uses position, color, and size to represent where a player took shots that season and to what percent accuracy. Since the heatmap consists of tessellating, hexagonal bins, it is easier to delineate between zones of shots. Each hexagon's color intensity is based on sort accuracy while the hexagon size is determined by the number of data points in that area. At a glance, viewers can easily see the number of shots made from specific zones and if those shots contributed to the team's success, making it a highly effective visualization. Multiple, linked visualizations often provide clearer insights into multidimensional data than do isolated views and may require analysts to zoom, scroll, pan, and navigate the view (Heer & Shneiderman, 2012).

The visual does not require excessive cognitive effort since it clearly shows the heatmap of where the shots were taken from on a basketball court. Unique insights are highlighted by showing the FG% and the total number of shots taken for that particular player. Finally, these sorts of charts are very commonplace among NBA analysts, so viewers will already be primed to understand the chart.

6.6. Feedback from Users

During our final presentation, we received positive feedback from our classmates regarding how our visualization was designed and how informative it was. Our visualization successfully reached our target audience, avid basketball fans who follow at least one team, since the feedback results were overwhelmingly positive. We believe this visualization expands on the cognitive abilities of our target audience (Mannheimer, 2022a, slide 80).

6.7. Reviewing Project Specifications

The goal of this project, which was to build a web-hosted, interactive prototype visualization demonstrating the core concept, was met. Having built our visualization in Tableau and making it interactive, we were able to deploy this on Tableau Public. Our visualization has multiple dashboards that clearly demonstrate our concept: to compare NBA players' salaries with player performance & other statistics across 38 NBA seasons. Furthermore, we met the specifications for valid data analysis as well. Our team upheld our graphical integrity by accurately visualizing relationships while presenting them in context. Every visualization is labeled clearly and carefully to avoid confusion and misunderstanding (Mannheimer, 2022b, slide 61).

6.8. Project Timeline

The project consisted of 5 stages: (1) Coming up with the concept pitch, (2) Proposing the visualization concept, (3) Design Ideation, (4) Design evaluation, (5) Concept production, and (6) Validation. We spent significant time validating the data presented in our visualization after our beta testing and presentation. A key course reference emphasizes the importance of validating and revising the visualization based on feedback (Cooper et al., 2014).

6.9. Conclusion to Initial Questions

The three graphs on our main dashboard, *Tracking Payroll and Performance (1985-2022)*, *Breakdown of Team Salary*, and *Player RAPTOR Score vs Salary*, cumulatively show that teams do not effectively spend their payroll. The correlation between RAPTOR score and a player's salary was highlighted and allowed users to examine how well a team navigates its salary cap to maximize performance. We also used player shot data and their per-game stats to give the user more information on each player to give context to their RAPTOR scores. Our data and graphs indicate that teams and players with large salaries do not necessarily guarantee championship wins or even exemplary performance for that season. When analyzing the *Player*

RAPTOR Score vs Salary graph for multiple seasons and teams, we see that it is not rare to find a negative correlation between salary and RAPTOR scores. This also indicates to NBA fans and players that a stellar NBA performance does not assure a large salary. The individual player dashboards clearly show the breakdown of a player's performance and help the audience determine the strengths and weaknesses of each player with respect to their position.

7. References

Cooper, A., Reimann, R., & Cronin, D. (2014). About Face: The Essentials of Interaction Design (4th ed.). John Wiley And Sons.

Jeffrey Heer and Ben Shneiderman. 2012. *Interactive Dynamics for Visual Analysis: A taxonomy of tools that support the fluent and flexible use of visualizations*. Queue 10, 2 (February 2012), 30–55. https://doi.org/10.1145/2133416.2146416

Mannheimer, Nathan. *Week 1 DATA 511 Introduction*. December. 2022a, https://canvas.uw.edu/courses/1579062/files/folder/Slides?preview=95497929. PowerPoint Presentation

Mannheimer, Nathan. *Week 2 DATA 511 Graphical Excellence and Integrity*. December. 2022b, https://canvas.uw.edu/courses/1579062/files/folder/Slides?preview=95497929. PowerPoint Presentation

Mannheimer, Nathan. *Week 7 DATA 511 Dashboards I*. December. 2022c, https://canvas.uw.edu/courses/1579062/files/folder/Slides?preview=95497929. PowerPoint Presentation

8. Appendix

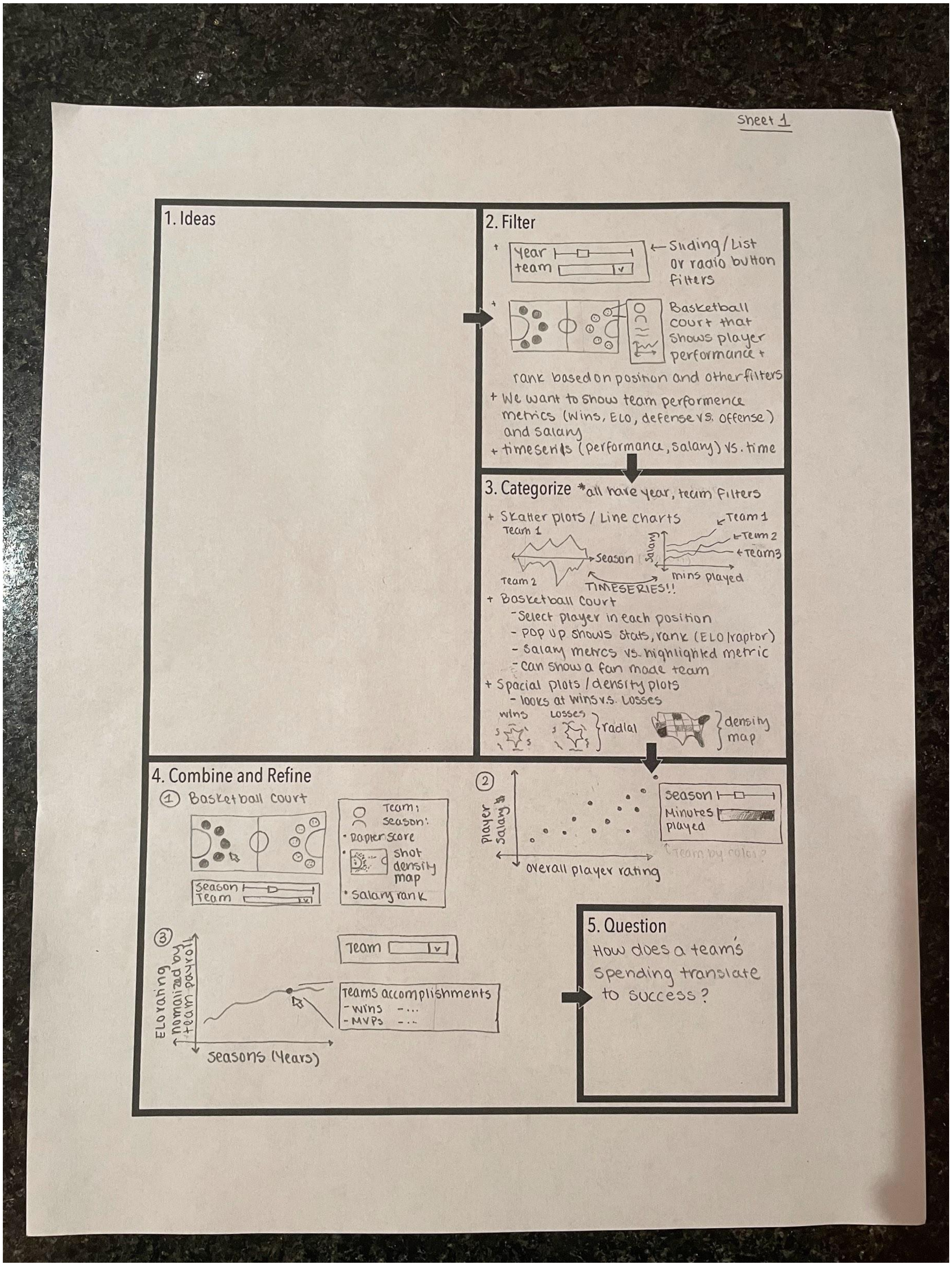


Figure (3)a. Design Ideation (Sheet 1)

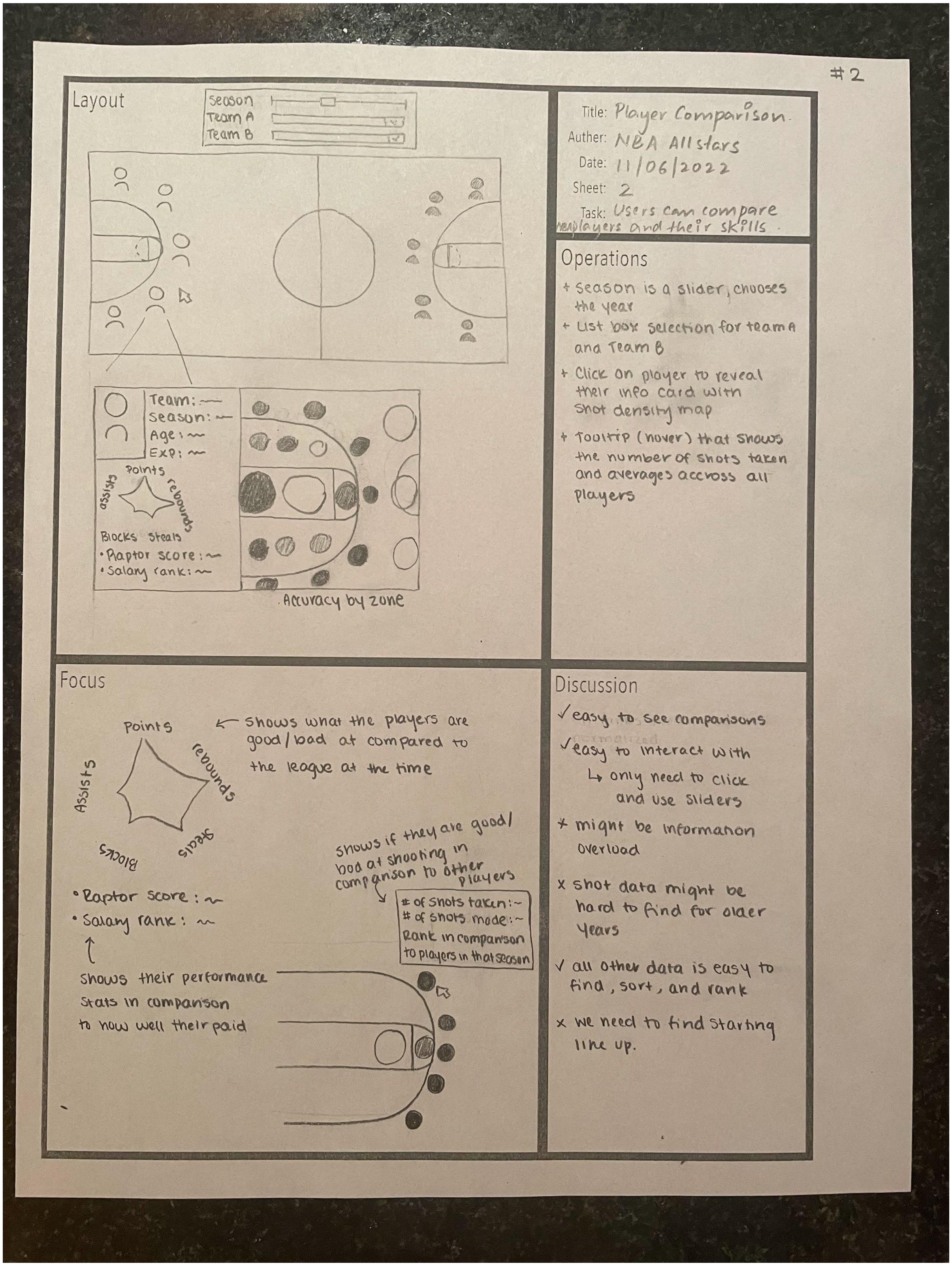


Figure (3)b. Design Ideation (Sheet 2)

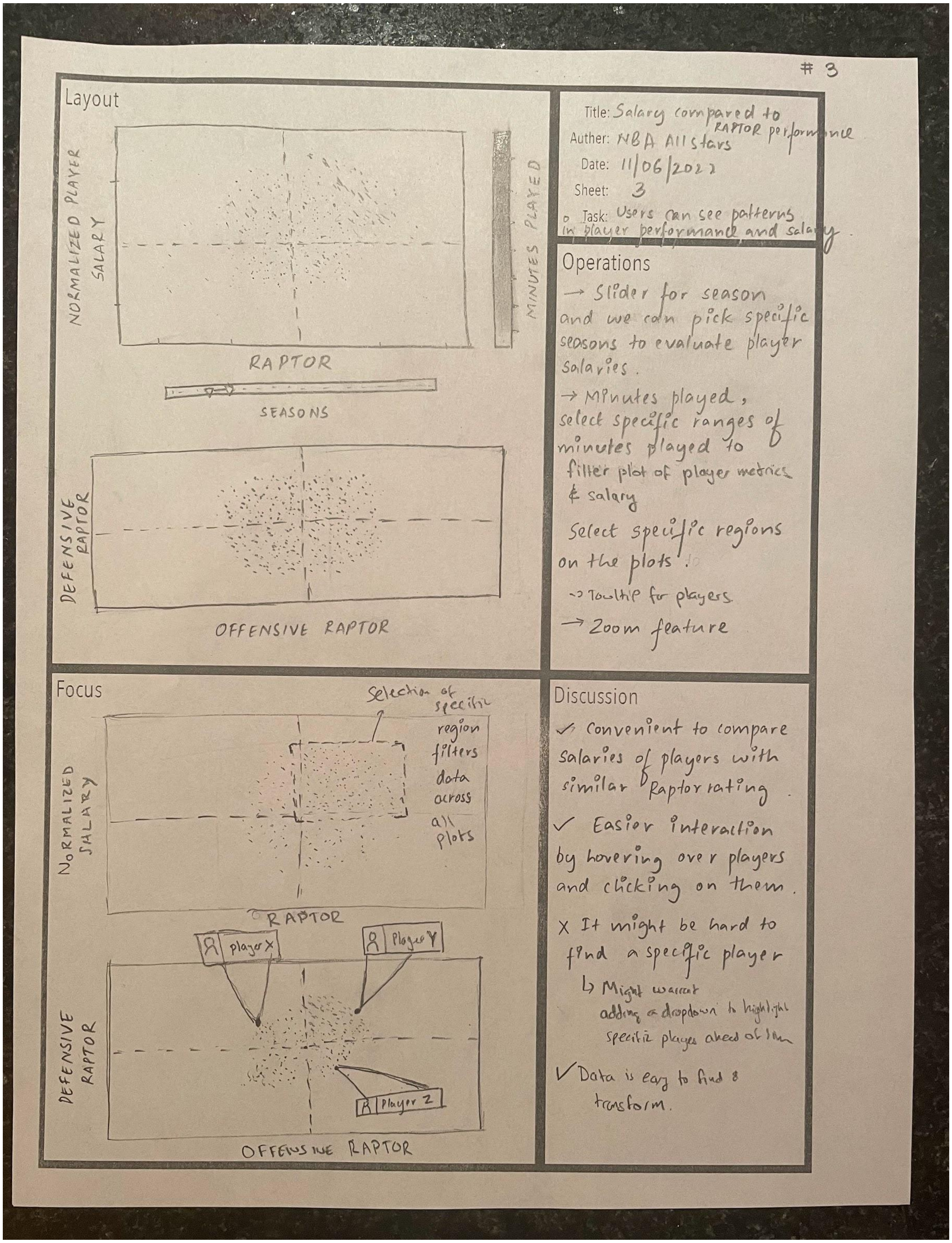


Figure (3)c. Design Ideation (Sheet 3)

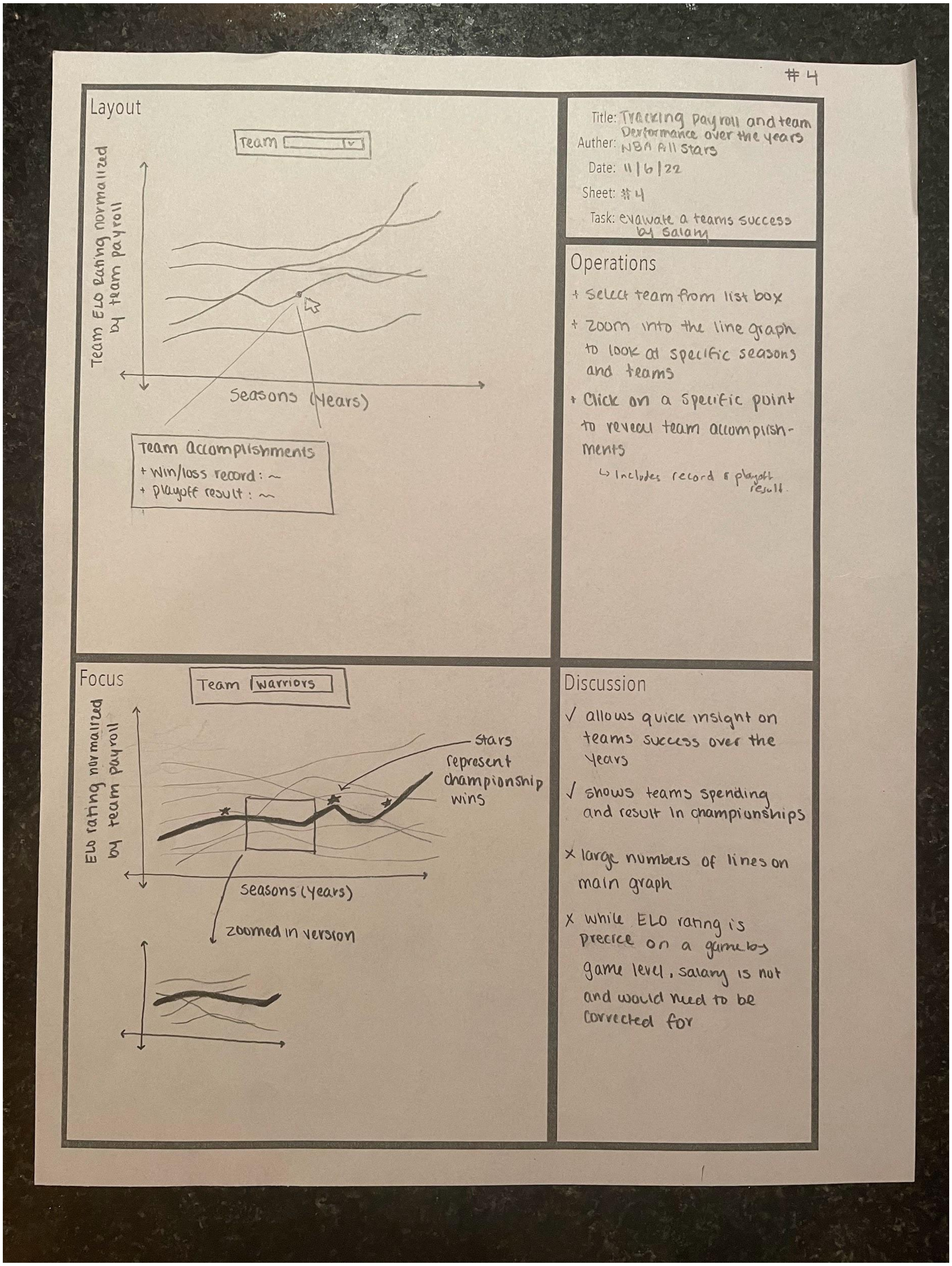


Figure (3)d: Design Ideation (Sheet 4)

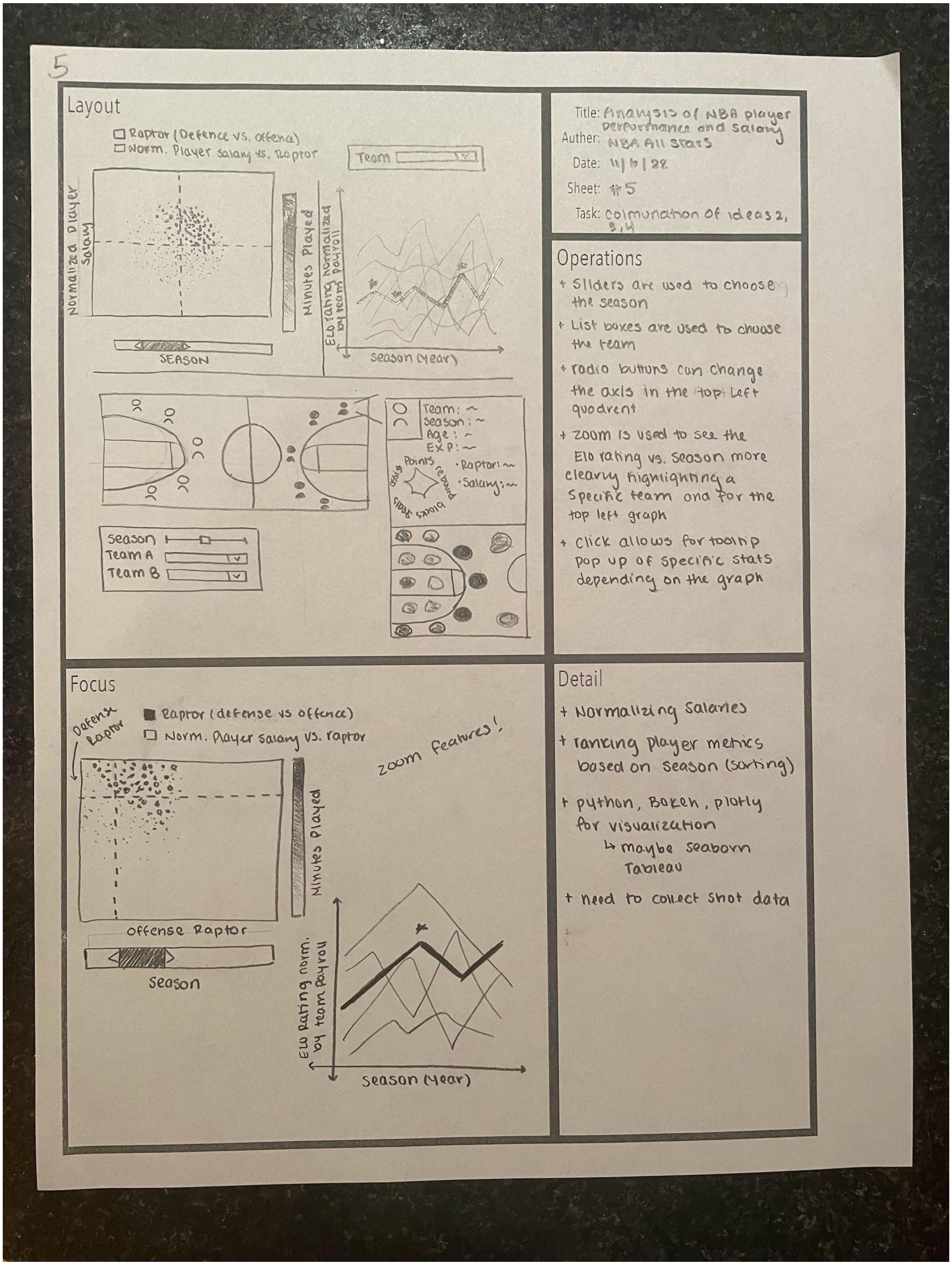


Figure (3)e: Design Ideation (Sheet 5)