

Davis Cohen

CSCI 3010 - Individual Project Proposal (revised)

Link to github repo:

[https://github.com/davis-cohen-3/NBA\\_analytics\\_website/upload/main](https://github.com/davis-cohen-3/NBA_analytics_website/upload/main)

### **(1) *What is my project?***

My proposal is to make a graphical user interface based application, where that covers NBA basketball analytics with several different features.

#### **Part 1 of project:**

First, the majority of this project will be dedicated to doing data analysis and modeling of individual player statistics (field goals made per game, free throw percentage, three-point percentage, rebounds, blocks, steals, etc.), efficiency, winning rates, etc - the goal will be to categorize each NBA player and further put players in groups with defining characteristics. After doing reports on individual players, I will make reports on teams, gms, owners and fanbases. The goal behind this subject of analysis will be to determine the characteristics of winning teams, mediocre teams and losing teams. Here are some examples on what my aim is: is there a positive relationship between a player's offensive efficiency and their three point accuracy? Do winning teams in the NBA tend to have an average height and length (arm length) that is greater than those of mediocre and losing teams? Etc... These are often questions that I think about, as I am what you would call a basketball savant - not only do I love playing basketball, but if I could do anything in the world it would be to be the head coach of a division 1 basketball team.

#### **Part 2 of project:**

The second part of this project will not deal with data analytics and modeling, rather it will be sort of a NBA basketball team simulator (kind of like a fantasy basketball app, but way more statistics based). Users will have a goal of designing teams that meet certain criteria. The reports done in the first part of this application are thought to guide the user to designing a winning team. I understand that a good majority of this project is supposed to deal with object oriented programming (classes, objects, inheritance, etc.), so to fulfill this objective I will have several different classes (team, personnel), under personnel will be several child classes (player, coach, gm, owner), under player will be more child classes (defensive player, two-way player, role player, three point specialist, ball handler, shot maker, star player & mvp-caliber player) \*design behind these player classes is subject to change\*.

All the players, gms, and owners will be real and current NBA players, gms and owners, and they will be categorized into each inherited class depending on the analysis done in part 1. Once teams are put together by the user, there will be a tailored report generated on the team made, and it will cover the potential for success that the team has. I plan on making graphs and models as well to go over the team generated by the user. Finally there will be a report card generated for the team.

The goal behind this component of the project is to allow people all across the world, and in particular people who are involved in the NBA, to have a tool designed for building successful teams.

Lastly, there will be a leaderboard on the application to go over the best generated teams.

## **(2) What technologies will I use?**

- **Python, pandas, and numPy** to fulfill part one of the project. I am pretty familiar with these technologies as I am taking intro to data science right now that depends on these technologies completely.
- **A CSV file for player stats:** link to file →  
<https://www.nbastuffer.com/2021-2022-nba-player-stats/>
- **Python** to fulfill part 2 of the project. I am familiar with Python, however not as familiar as I am with C++.
- **Django** to make this a secure website. I am not exactly sure what it will take to develop a backend, and whether I will use a local server or the cloud. I am completely unfamiliar with Django
- Note: I am not planning on using a database yet, for all the player and team data, I will retrieve the numbers from the imported csv files I use. The only reason I would use a database would be to keep track of the users and the teams they create.

## **(3) Essentials to this project:**

- I need to make my analysis and models with Python, pandas, and numPy frameworks, otherwise this project will not transpire into anything. Is there a backup plan for this? NO! But I figure this part will be not necessarily trivial, but definitely manageable.
- I need to make a simulator (Python) in order for this project to include user activity and user input. There again is no backup plan for this component. If I don't have these then my project will simply be a bunch of reports and obviously I wanted to make this a object oriented project, so this part of the project is a MUST.

- There definitely are other essentials, such as managing the website with a backend, which might just entail a bunch of post and get apis to make the website functional.

### **Resources:**

I need a lot of data on individual NBA players and NBA teams over the past 5 years. Since the NBA is always changing, I want to base my reports off of the past 5-7 years, as the league has definitely transformed since 2015. I found this excel sheet that holds all the individual player data, I can export this as a csv file (link: <https://www.nbastuffer.com/2021-2022-nba-player-stats/>) , as well as team data sheets (link: <https://www.nbastuffer.com/2021-2022-nba-team-stats/> ). These two data sheets provide me all the data I really need when it comes to measuring players and teams.

As far as art work goes, I really don't think I will need art, however I have friends who might be able to help me with some graphics if I think they might be a cool addition to the project.

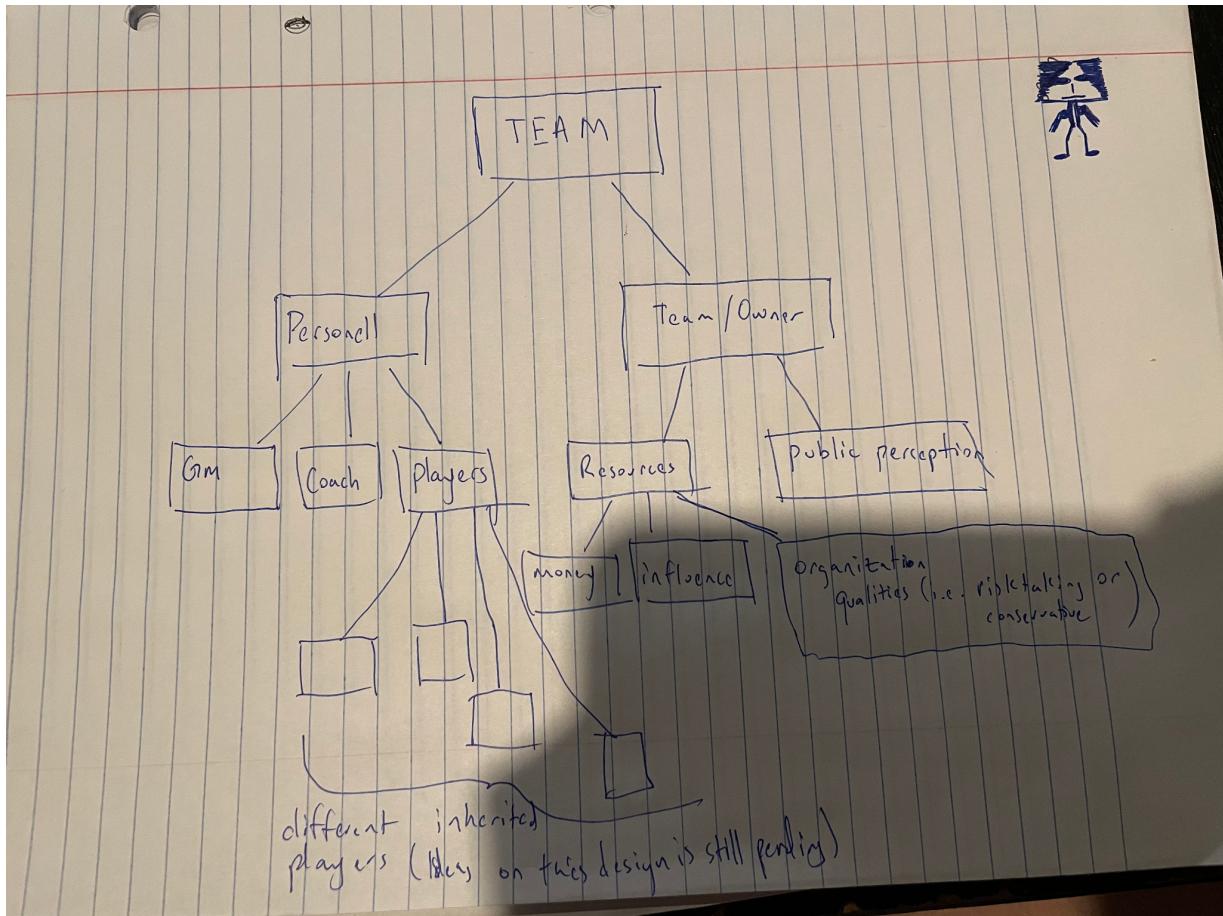
### **(4) Architecture Proposal:**

I will have both a frontend and a backend. There will be three different pages: 1. First pages will be divided into two pages where each page covers different types of reports. There will be a team success report page. An individual player success page. For each of these two pages there will be a menu listing all of the different types of analyses and models for either players or teams. The second page will be the simulator part of this application. There will be a submit team button at the bottom. A search bar for players, gms, coaches and owners. Lastly there will be a table which must be filled completely in order to submit the team. Once the submit button is clicked, a report will be generated based on some type of api.

I briefly touched on this earlier, but here is my plan for integrating object models:

- Team is the main class which will consist of personnel and owner(franchise)
- When a team is instantiated it will have the right amount of players, gm, etc. assigned to it
- The salary cap for each team will be dependent on the owner chosen.
- I will run a program most likely through the main file where the user is prompted to select owner, players, etc.

- The owner selected will have its own pre-defined resources, money (salary cap), influence, perception, etc.. These might just be characteristics of the owner/franchise instead of individual classes.
- Where inheritance comes into play is in the player class. Where there will be different types of players (superstars, 3-point specialists, lockdown defenders, etc.) Design on these classes is still in the works.



Design Patterns: I am going to use either the Singleton pattern or the factory pattern. I am not really sure what the benefits for my particular project for each of these patterns are, but this is something I will discuss in meetings.

#### Database design:

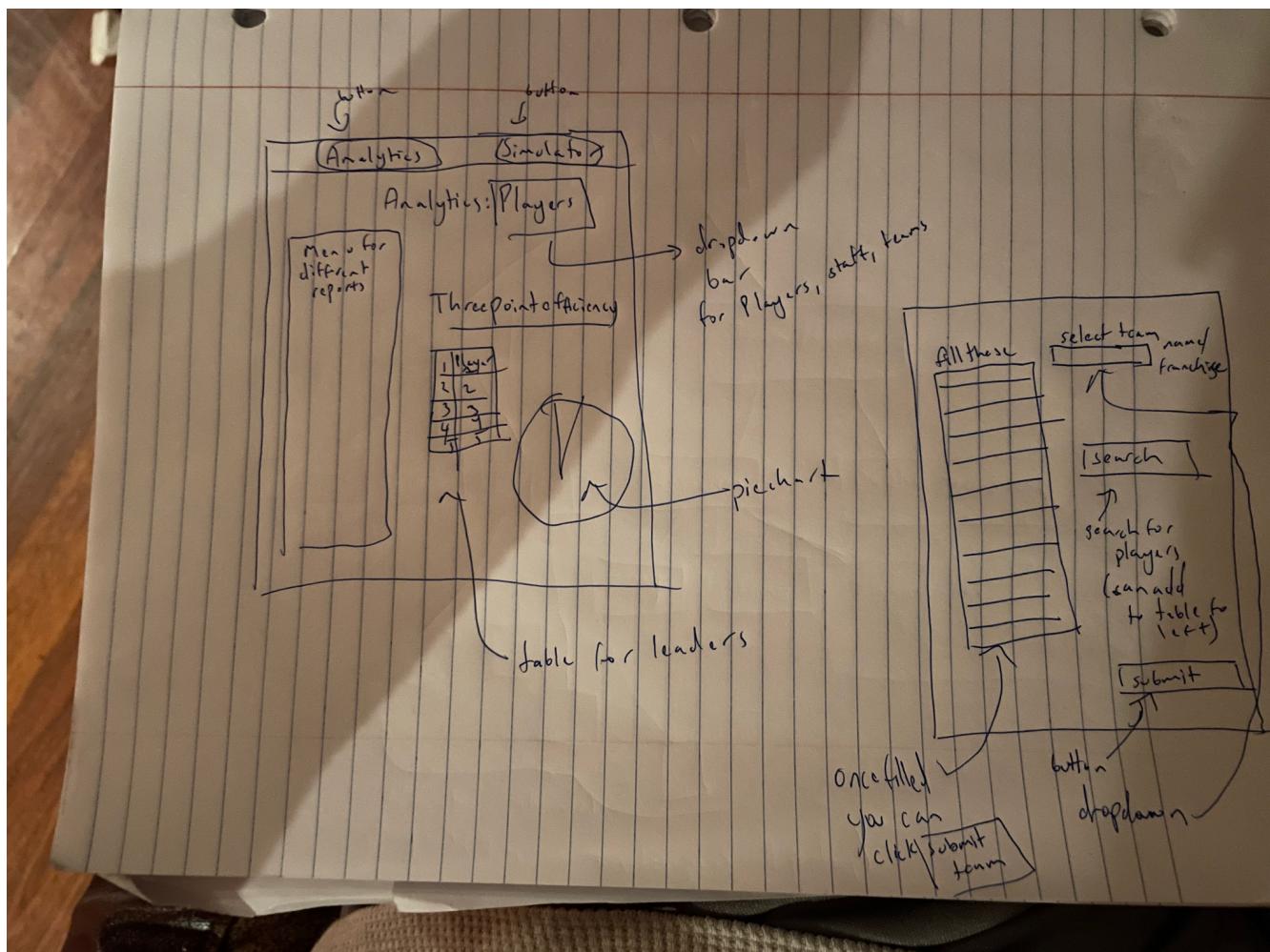
I will only need a database to keep track of the users of the website and the teams that they generate. All the other data on players and teams will be held in large csv files, which I will make dataframes of in python.

## (5) Website Plan:

I've discussed above the page format I will have. Here are a few diagrams of how I plan it will look.

### Pages:

- Analytics page (multiple pages : players, coaches, owners/franchise)
- Simulator Page
- Scoreboard page (this might need a database)



## (6) Schedule:

Feb 27:

- Have every technology pinned down
- Get all the data needed
- I have all the knowledge I need to make the reports, I just need to get all the data I need
- I also might need to do some more research on getting the first page of the gui up

### March 11:

- Get the majority of my analyses and models done
- Again I have everything I need to do this

### April 8:

- Implement the backend
- Get the simulator working to the point where it's functional, but not yet complete

### April 19th:

- Make sure that all the reports are generated exactly as they should be in the simulator part of this project
- Start setting up a testing environment for this simulator
- Continue to perfect the simulator

### April 28th:

- At this point everything will be good to go
- Launch the website or gui for public use

### **(7) How I will stay on track with class material:**