

### **Project Milestone 3**

Completed features: Getting individual player and team data and navigation between those

What worked: Everything we had planned worked even though it didn't look so good

What issues were faced: The Ajax calls were really hard to figure out but now that we have those working, we can build the rest of the app with no problem.

TA Questions:

Q: What are your full features?

A: Home page + animations, player stats and team stats, populated database, and working node API

Q: Where are you getting the data from?

A: Sports-reference API and use python to stuff it in database

Q: Are you running CRON to get updated data?

A: We have the framework to start grabbing data but for now we have data for the past 10 years to get the data and we will be pulling recent things soon

Q: Where do you go from here?

A: Make it look nicer, provide more in-depth information, make graphs for players and teams, and using elo or trueskill, better navigation (search that searches), and have index of players and teams

Q: Is this working for all the sports?

A: Only NBA right now but easy to transfer to the different sports, only difference is stats

Q: What are you using for your backend and database?

A: Postgres and populating it with python hosted in AWS, Node API that calls back to server

Final thoughts: Demo looks good and full pages, but wants to see it for all sports ASAP and wants an elo system before the final demo. Polish UI and get the demo looking good before final. Your UI needs more info for teams and players. He wants graphs and cool info.

Individual contributions:

Enrico: I made the Node API and got the ajax calls to talk to the frontend. Helped with functional React code for populating fields and reloading/routing the site.

Abdul: Created hover animations and clickable cards for navigation on the main page. Created a search bar on the team page in order to present different team information. Various front end updates to the different pages, still being fleshed out.

Henry: Database structure, database population, and SQL queries

AJ: Worked on stylizing the frontend features through utilizing tools from React and Bootstrap

Evan: Started graph infrastructure for player and team statistics.