

Project Abstract
CMPE 138/180B : Database Systems
Project Group 17

Project Title:

NCAA Basketball Analytics: Insights into Game Trends and Performance Metrics

[Dataset Link](#)

Group Members:

1. Zach Kuo
2. Phuong Duy Lam Nguyen
3. Kunal Keshav Singh Sahni
4. Sonali Lonkar

Abstract:

This project leverages the **NCAA Basketball dataset** available through Google BigQuery's public repository to **build a scalable analytical API** that delivers real-time insights into game statistics, team performance, and player metrics. Our primary focus remains on designing and optimizing the data layer—including an effective database schema, indexing strategies, and query tuning—but we extend our work by developing an API that serves as a direct interface for accessing these insights.

The API will expose endpoints to support key functionalities:

- **Game Statistics and Trends:** Users can query historical game results—including scores, dates, and participating teams—and analyze trends over time (**e.g., average points per game, win-loss records**).
- **Team Performance Metrics:** Endpoints will provide access to detailed team data such as **offensive and defensive efficiency**, points per possession, and other advanced performance metrics, facilitating cross-season comparisons or head-to-head analysis.
- **Player Analytics:** Individual player statistics (**points, rebounds, assists, etc.**) will be retrievable through API calls, enabling granular analysis of player performance.
- **Player to Player Comparison :** Performance comparison between players based on their metrics enabling a deeper understanding of individual contributions, strengths and weaknesses. (**Key metrics for comparison - scoring efficiency, playmaking and passing, defensive impact, overall efficiency**)

- **Season Winner Predictive Analysis:** Implement predictive modeling to forecast the winning or losing outcomes of teams in upcoming seasons by training machine learning models on historical data (**team performance metrics, player statistics, game outcomes, etc.**) from previous seasons and testing the predictions on future season data.

The underlying data model is carefully designed to normalize and represent relationships among games, teams, and players while also incorporating effective indexing (e.g., B+ trees and hash indexes) and query optimization strategies to minimize I/O costs and enhance performance. By building a single point RESTful API on this optimized data layer, we aim to not only demonstrate robust database design but also enable seamless integration with front-end applications, reporting tools, or other consumption channels.

Tentative Timeline

Week Number	Date	Details
1	Apr 8, 2025	<ul style="list-style-type: none"> • Project discussion and dataset exploration. • Establish project goals, roles, and initial design considerations.
2	Apr 15, 2025	<ul style="list-style-type: none"> • Finalize project requirements and design an initial database schema for game, team, and player data. • Outline the API specifications and endpoints.
3	Apr 22, 2025	<ul style="list-style-type: none"> • Implement data ingestion routines to load and normalize the NCAA Basketball dataset into the database. • Begin developing core SQL queries for key analytical features.
4	Apr 29, 2025	<ul style="list-style-type: none"> • Develop and test SQL queries for: <ul style="list-style-type: none"> • Historical game statistics and trends • Team performance metrics • Player analytics • Initiate basic API development using RESTful principles.
5	May 6, 2025	<ul style="list-style-type: none"> • Focus on query optimization and indexing strategies. • Integrate the data layer with the API, ensuring that endpoints efficiently serve query results.

		<ul style="list-style-type: none">• Perform performance analysis and fine-tune API response times.
6	May 8, 2025	<ul style="list-style-type: none">• Final integration and comprehensive testing of the API.• Prepare the final report, presentation slides, and demo of the API functionality.• Submit the project report and present an 8-minute live demo highlighting our database design, API endpoints, and performance optimizations.