Chase Rodie

04/14/24

Final Project Design

1. The goal of this NBA Database software system is to create a comprehensive platform for storing, retrieving, and analyzing a wide array of NBA-related data. This encompasses player statistics, team information, game results, and historical records. The system aims to cater to both casual fans and analysts, providing intuitive interfaces for browsing data and conducting advanced queries. It serves as a centralized repository, meeting the needs of NBA enthusiasts, from quick insights for casual followers to in-depth research tools for analysts. This will be limited to the 2023-2024 mainly because trying to incorporate anything more than that can get way over complicated.
2. For the NBA Database, I've selected a tech stack comprising various tools and technologies to facilitate development, management, and analysis of NBA data. At its core, the system utilizes Python3 for scripting and data manipulation, with the psycopg2 library enabling seamless interaction with the PostgreSQL database management system. For database administration and visualization, I'll leverage pgAdmin 4, serving as a graphical administration tool for managing PostgreSQL databases. The tech stack also includes integration with external data sources, such as basketball reference datasets, to populate the database with comprehensive and up-to-date information. Python scripts will handle data import and manipulation, making sure that the data is accurate and adherence to the schema.
3. The infrastructure planned for hosting and running the NBA Database system primarily relies on a local development environment. Utilizing tools like VSCode for Python scripting and pgAdmin 4 for PostgreSQL database management offers flexibility, accessibility, and complete control over the development environment. By keeping everything local, there's no need for external hosting services, streamlining the deployment process and allowing focused efforts on enhancing the functionality my Database.
4. The primary outside data source for this project is basketball reference datasets, provided in CSV format. These datasets contain comprehensive NBA information, including player statistics, team data, game results, and more. The plan for incorporating this data involves using pgAdmin 4 to import the CSV files into PostgreSQL tables. Python scripts, using libraries like pandas, will facilitate data manipulation and integration tasks, ensuring the data is properly structured and aligned with the database schema.
5. A diagram of a basketball game

   Description automatically generated

6. ---------------------------------------------

|NBA Database:

-playerStats: PlayerStatsService

-teamStandings: TeamStandingsService

-userFavorites: UserFavoritesService

---------------------------------------------

|PlayerStatsService:

+getPlayerStats(): DataFrame

---------------------------------------------

|TeamStandingsService:

+getTeamStandings(): DataFrame

---------------------------------------------

|UserFavoritesService:

+addUserFavorite():void

+getUserFavorites(): DataFrame

7. The frontend interacts with the backend via RESTful API endpoints provided by Flask (Python3 script). The backend communicates with the PostgreSQL database to retrieve and manipulate data based on user requests via the CLI. Additionally, the backend interacts with external data sources, such as basketball reference datasets, for importing and integrating additional data into the database. The frontend utilizes this data to display information to the user and allows interactions such as querying and filtering of the results.

Top of Form

Bottom of Form