Project Report: NBA Sports Betting Assistant

1.Please list out changes in the directions of your project if the final project is different from your original proposal (based on your stage 1 proposal submission).

- Initially, we planned to include a feature for simulating betting outcomes using blackbox ML algorithms, but due to time constraints and complexity, we decided to work on it in the future. We ended up using a probabilistic model for our predictions.
- We ended up not creating visualizations but with concise stored procedures to give Users a simple way of reading important information to place their bets.
- Real-time updates from external APIs were excluded to focus on historical data integration.
- We initially proposed integrating multiple APIs but later just used Kaggle datasets for consistency and to ease data processing.
- Biggest change: we moved from NFL to NBA

2.Discuss what you think your application achieved or failed to achieve regarding its usefulness.

- Achievements:
  - The application successfully aggregates historical NBA data and provides insights for informed betting decisions.
  - Users can analyze team performance, track their betting history
  - CRUD operations on betting history work seamlessly, allowing users to maintain accurate records.
- Failures:
  - The lack of real-time data integration limits the app's applicability for ongoing games.

3. Discuss if you change the schema or source of the data for your application

- Schema Updates:
  - UserBets Table: We only changed it in stage 2 when we made it a 3-relational table
- Reasons for Changes:
  - Storing betting information within UserInfo would lead to redundancy, as it would duplicate user information for each bet.

4. Discuss what you change to your ER diagram and/or your table implementations. What are some differences between the original design and the final design? Why? What do you think is a more suitable design?

- Our main change was turning UserBets into a 3-relational table. This helped with our normalization.

---

5. Discuss what functionalities you added or removed. Why?

- Added:
    - Filtering games by date or team for targeted analysis.
    - Calculation of win percentages for both home and away games.
- Removed:
    - Simulation of betting outcomes due to complexity.
    - Real-time API integration to focus on refining historical data functionalities.
- Reasons for Changes:
    - Time constraints and technical complexity drove the removal of certain features.
    - We prioritized core features like CRUD operations and historical trend analysis.

---

6. Explain how you think your advanced database programs complement your application.

The advanced SQL queries complement the application by:

- Providing actionable insights, such as win percentages and average points per game.
- Allowing real-time computation of metrics like team rankings based on wins.
- Optimizing performance through tailored indices, ensuring smooth data retrieval.

---

7. Each team member should describe one technical challenge that the team encountered. This should be sufficiently detailed such that another future team could use this as helpful advice if they were to start a similar project or where to maintain your project.

Challenge 1: GCP Issues (Aarush)

- Issue:Data formatting lead to tables with mismatching values and unable to make foreign keys
- Solution: Formatted and uploaded data correctly-make sure to remove \r from strings after uploading from excel

Challenge 2: Data Cleaning (Abhay)

- Issue: Inconsistent formatting of team names across datasets.
- Solution: Created a mapping script in Python to standardize team names during data ingestion.

Challenge 3: Using GCP in Flask Code (Nishk)

- Issue: Difficulty in dynamically linking GCP queries with the Flask front end.
- Solution: Added IP as a registered user for the database to ensure all queries would be usable on the Google Cloud Project Database

Challenge 4: Stored Procedures (Ethan)

- Issue: Consistent Issues with implementing and using stored procedures
- Solution: Used mutli=true flags in stored procedures when queried on the database to execute multiple statements sequentially

---

8. Are there other things that changed comparing the final application with the original proposal?

- None other than described earlier

---

9. Describe future work that you think, other than the interface, that the application can improve on

- Real-Time Enhancements:

    - Integration of real-time data sources like SportsData.io for live updates.
    - Implementation of betting outcome simulations.
- Data Insights:
    - Development of machine learning models to predict game outcomes and recommend bets.
- User Experience:
    - Improved interactive visualizations with advanced filtering options.
    - Mobile app integration for on-the-go accessibility.

---

10. Describe the final division of labor and how well you managed teamwork.

Team Member	Responsibility	Remarks
Nishk	Frontend (UI/UX)	Successfully implemented and tested UI workflows.
Ethan	Backend and API Integration	Simplified data processing pipelines.
Abhay	Data Analysis/SQL	Developed efficient advanced queries.
Aarush	Database Design and Deployment	Ensured schema normalization and optimal indexing.

Team Management:

- We checked in weekly during after class to make sure we were on track
- Distributed work based on who was most experienced and comfortable