Final Project Report

Links

Demo video: https://youtu.be/gCNFDhZKoQ8

http://34.56.8.148 - IP for application running on GCP

Overview

The Sports Trivia App is an web application that provides trivia games featuring football questions for sports fans to test their knowledge of football events. The app compiles trivia questions based on official match data while also allowing users to contribute their own questions that can be added to the question pool. With a specific focus on the NFL and CFB landscape, the Sports Trivia App is an interactive educational tool to add value to the football community.

Application Walkthrough

Users start by authenticating at the login page. Once authenticated, they proceed to the home page where they are presented with the options of taking an NFL or CFB trivia quiz. Each quiz consists of 10 questions, and the question content covers a wide range of data such as player accolades, in-season awards, and championship winners. The questions are randomly selected from a pool of official and user submitted questions, and each is of varying degrees of difficulty. Upon completion, the user is provided their score and the leaderboard is shown, displaying the best scores from the community.

Directional Pivots and Functionality

Our team made several strategic adjustments to the project's direction and feature set in response to both practical constraints and a refined vision for the application's core experience. In our initial Stage 1 proposal, we outlined an ambitious set of features designed to maximize user engagement and customization. However, as the semester progressed, we recognized the need to prioritize essential functionality and user experience over breadth of features. This led us to streamline the application, focusing on delivering a robust and enjoyable trivia experience for football fanatics.

Several features from our original plan were either removed or deferred for future development:

- User-Created Quizzes:
    - We initially intended to allow users to create and upload entire trivia quizzes for others to attempt. We decided this feature was misaligned with our goal of providing a seamless and accessible trivia experience. We wanted to minimize the effort required from users and instead offer a curated pool of thousands of unique questions, ensuring consistent quality and variety
- Editing User-Submitted Questions:
    - The original design included functionality for users to update their own submitted questions. We removed this feature to streamline the review process. All user-submitted questions are now tagged with a status attribute and reviewed by the admin team. Approved questions are added to the main question pool, while

unapproved ones are deleted. Users wishing to revise a rejected question can simply submit a new version.

- Searching for Other Players' High Scores:
    - We considered implementing a search bar for users to look up other players' high scores by username. After evaluating the complexity and the potential for more meaningful social features, we decided to defer this functionality. We concluded that a friends system or broader social features would be better developed as a cohesive package in future iterations, allowing us to concentrate on the core trivia gameplay for this release.

These pivots reflect a deliberate shift toward delivering a high-quality, focused trivia platform.

Application Achievements and Usefulness

The Sports Trivia App successfully fulfills its core mission of providing an engaging and educational platform for diehard football fans to test and expand their knowledge of NFL and CFB history. By combining a curated pool of official questions with user-submitted content, the app ensures a diverse and continually evolving set of trivia challenges. This approach not only maintains the relevance and freshness of the question bank but also encourages community participation, making users feel invested in the quality of the quizzes.

From a usability standpoint, the application achieves a smooth and intuitive user experience. The authentication process is straightforward, and users are quickly guided to their choice of NFL or CFB trivia quizzes. The random selection of questions ensures that each quiz attempt is unique, increasing replay value and preventing memorization. The inclusion of varying difficulty levels allows both casual fans and hardcore enthusiasts to find a suitable challenge.

Additionally, the leaderboard feature adds a competitive dimension, motivating users to improve their scores and engage with the app repeatedly. By displaying top community scores, the app fosters a sense of achievement and community among users.

Schema and UML Diagram Design

So far from the last checkpoint, the source of the data is enough for us to provide sufficient and high quality questions. For our ER/UML diagram, we changed it greatly since we mix the definition of ER and UML diagram, so we edit it to only use UML diagrams. After we made this change to our UML diagram, we included a difficult attribution in our UserSubmission entity, which enabled us to union two dataframes to make combined questions. We introduce this combined question as a new design for our application that can enhance the competitive dimension of the app and create a more engaging trivia experience for users.

Advanced Database Programs Integration

The advanced database program complements our application by making most of what is done simpler from the users end. The code looks complex, but for example the transactions help make our database simpler. Instead of creating many new data frames for each specific leaderboard part of the code we can use a query to show the user the union of different data

frames. This simplifies the database and complements our application because users can easily see the combination of different data frames. Another example is with the trigger. This trigger is simple, but it is extremely effective because it ensures that questions can't be spammed and are phrased correctly. If the questions aren't framed correctly this can be bad for users who are taking the trivia and lead to a poor experience.

Technical Challenges and Lessons Learned

Kiernan: One technical challenge that the team faced was navigating the Google Cloud environment. We all had no experience hosting a database on this platform, so it took some time for us all to become familiar with the Cloud SQL interface, access controls, and the various steps involved in creating and managing a database instance. What helped us through this process was documenting our setup and procedures so that the whole team could get up to speed quickly and have a reference if we ever forgot something. Communication is also very important. We worked through these issues on a Discord call together to help each other out and stay on the same page.

Ryan: A technical challenge that the team encountered was related to making the UML design. Originally this design was viewed as an extra part of the project, so at first not too much time was dedicated from all of us. As we worked on the project, we began to notice how essential the diagram was for understanding our database. This diagram and part of the project became one of our most referred to pieces of work when creating the app in the end and was extremely helpful once it was made correctly.

Zihao: One technical challenge that we encountered was managing user-submitted questions within the existing trivia question pool. Initially, we tried to give users the ability to update their own submitted questions. However, we realized that this would make the question pool harder to manage and would increase the complexity of our code. To address this, we updated the feature by adding a status tag for each submitted question, allowing us to review user submissions and ensure that the database remained organized and consistent. This experience taught us the importance of planning for database flexibility early in the design phase, especially when new types of content might be added later.

Shiyuan: One technical challenge that the team encountered was managing authentication and protecting user data securely. Our original design was to store the user password directly in the database to access it. However, after some research, we realized how important it was to properly hash and store passwords instead of saving them in plain text, so we hash passwords before storing user credentials in the database. This process helps us realize the importance of data security.

Future Improvement Opportunities

While the current version of the Sports Trivia App delivers a solid foundation for football trivia enthusiasts, there are several features we envision for future development. These improvements aim to deepen user engagement, foster a stronger community, and provide richer feedback and analytics for players.

Potential Features:

- Endless Trivia Mode:
    - Endless trivia would allow users to answer questions continuously until they make a mistake. This feature would offer a new challenge for competitive users and encourage longer play sessions.
- Enhanced Social Interaction:
    - A stronger community could be built through features such as the ability to add friends, view their trivia high scores, and perhaps even compete directly against them. These social elements would make the app more interactive and engaging.
- Streak-Based Scoring:
    - We envision a more advanced scoring system that rewards players for consecutive correct answers, similar to the answer streaks popularized by platforms such as Kahoot. This would incentivize the combination of accuracy and answering speed, making gameplay more dynamic.
- Advanced Player Analytics:
    - Providing users with deeper insights into their performance is another area for growth. Planned analytics include displaying each player's average trivia score by category and mode, as well as tracking their best answer streaks. These features would help users gauge improvement and set personal goals.

Teamwork and Division of Labor

Our team took a very collaborative approach throughout each of the project stages. We would all join a Discord call, create a Google Doc for meeting notes and to build out stage requirements, and then openly share our ideas for how to accomplish our tasks. For bulkier stages such as Stage 3, we created the schema and tables together, then split off to each complete an advanced query, and finally rejoined for index testing. We managed teamwork effectively by staying in weekly communication through Discord.