Colin Devenney
Snigdha Gupta
Hamza Husain
Alex Murphy

SportsNow Application Final Report

1. The main change in direction was the idea of personalization for specific users. Originally, our idea was to keep a comprehensive database for the players and teams for specific sports that a user would like to see. However in the interest of preserving the original purpose of the application, which was statistics analysis, we focused more on providing stats for all players in the database. We specifically prioritized basketball statistics because that is what we all felt most passionate about. Furthermore, we didn't add the functionality of customizing the dashboard as we originally discussed in our proposal.

2. In terms of what our project succeeded in, we were able to successfully create an application for viewing stats of NBA players and returning two intriguing statistics about specific players and teams in the NBA. Specifically, finding the number of home wins a team has and finding the average ratings of players for each height range (in cm). Our current project could serve as a starting point for implementing more queries to find advanced statistics on players and teams. We failed to offer customizability and personalization for users.

3. No schema changes and we added some more data for player stats from online datasets.

4. There were no significant changes to our database design because we were able to parse specific data that we wanted and put it in a format that an online converter was able to easily create the tables for. A more suitable design would have been to structure our tables exactly like the datasets we used but for our purposes it was enough to keep our schema and alter the dataset according to our needs.

5. One functionality that we were not fully able to implement was the personalized data, we wanted to ensure that the core of the app, which was finding statistics, was preserved. In terms of finding statistics, our app succeeded in its functionality purposes. Another functionality we didn’t implement was betting odds but that would have been very complicated to implement so we decided against it.

6. For the advanced database program, we decided to implement our advanced queries (finding the average rating by height and number of home wins per team) in our stored procedure and return them as tables. Then we loop through each of the new tables and check our parameters that we pass in and only return the teams/heights that have greater values). This is relevant to the application for looking at statistics of players and teams. Our trigger simply ensures that the max rating of players is no greater than 100 to ensure that it is on a scale of 1-100. Hence our advanced database program contributes to the overall purpose of the application, which is to provide advanced statistics to users.

7.

Colin technical problem: One issue I found was trying to find enough data and scaling it to be accepted by our database. I solved this issue by using online CSV to SQL converters which made inserting the data into the database much easier. For future reference, I would make our schema more flexible to accept known dataset formats. The stored procedure also caused some problems as we thought about how to implement our advanced queries in the stored procedure.

Alex technical problem: One technical issue I faced was using SQL queries in our Flask backend. Reading the documentation and playing around with some code for the sql library we had to use made it possible for us to write high quality queries in SQL. After gaining that knowledge I successfully communicated about my findings with my teammates and enabled everyone to work on the backend queries successfully and implement them with ease.

Snigdha technical problem: One issue I found was smoothly integrating Google Cloud Platform and our MySQL Database. We also faced some difficulty in connecting our application to a MySQL instance on GCP. I also personally faced several issues with setting up GCP and had to take help from my teammates. So I believe, this issue could easily be solved by having a much better understanding of GCP and taking advantage of the office hours

Hamza technical problem: Technical problems that I faced were mostly related to frontend as it was a new area for me. Connectivity issues like connecting the server and database to the application was also slightly confusing and lead to some issues. Taking help from teammates in these situations was really helpful.

8. The design sketched up in our proposal is different from our final design, there was just one page for the final design.

9. We could add more triggers to improve functionality and ensure data is sound (in terms of relevancy and making sure impossible data is not included). Additionally, if the CRUD + search operations were kept in stored procedures, then our project would be more secure against script injection attacks. Furthermore, we could make the front end a bit prettier.

10.

Colin: Added data to SQL database, wrote two advanced queries, made stored procedure and trigger, helped in connecting back end nodeJS server to SQL database.

Alex: Setup GCP SQL and created and designed the initial database. Helped out with both frontend and backend to make sure API calls function and React site is working properly.

Snigdha: developed the base frontend, connected the frontend to the backend and implemented the SQL queries in the frontend.

Hamza: developed the frontend and connected it to the backend by using appropriate libraries and connected to the custom server and it’s API

Overall, we were able to work together pretty well with a few small communication problems in terms of specific deadlines. However, we were able to come together and create a final product we were all satisfied with. Everyone contributed equally and we were able to efficiently produce our desired product.