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

Our final project has the same key functionality as the original proposal. There are no significant changes in the direction that the project took in terms of project goals.

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

The application was able to largely achieve its main purpose in informing new football fans which games would be the best to watch for them. It puts together a variety of metrics for making such recommendations and compiles them for the viewer to see and understand. One area for additional usefulness would be to incorporate more statistics in our data in our decision algorithms, in order to give a more precise result.

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

The source of our data stayed the same throughout our application. In terms of our schema, we made some ad-hoc changes to our UserResult table with our foreign Id + “UserId”. The reason for it was our database was having issues with the user input being deleted would cause errors within our application since we had an “on delete cascade” action that would timeout our database. To respond to this we change the on delete actions to be a set to no action fixing our problem.

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 ER and table implementations are for the most part the same as our original design, however, the differences made were based on ad-hoc changes on handling foreign key id, to handle on delete errors causing our database to timeout.

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

One functionality that we added was a page for administrators to make changes to data in our database. Given the fast-paced nature of the NFL, trades between teams are frequent, and the data needed to be updated accordingly. Thus, we added a page where certain users could change player and team information to account for such changes.

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

Our advanced database programs add additional metrics which inform the user of key games or teams to watch. For example, our stored procedure creates a unique rating out of 5 stars for every NFL team, so the user would have an idea of the relative strength of each team. Furthermore, our trigger finds a matchup of a user's favorite team with the defending super bowl champion or runner-up. This allows the system to recommend an additional game to the user that is exciting and impactful.

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.

Mario 🌹 one challenge we faced was during development we had to combine our front end (React.js) with our backend framework flask into one coherent repository. To do this we created a backend directory to house our flask backend then could implement our API routes as well as set our local host as a proxy for the frontend. For the frontend, we just used our src directory to house it for its components to have a SPA and use the proxy used by the backend to send and receive data.

Sathvik: One challenge we faced during development was the advanced queries we wrote needed constant changing due to the bounds of what we were applying it. That took up a lot of time so for future teams, I recommend using either more simple queries while still maintaining the requirements or writing a query that they know they will use later for sure.

Vivek: One technical challenge we encountered was that elements of our schema design did not incorporate well into our backend and advanced elements. For example, one of our on delete settings for a foreign key caused our backend to crash multiple times when our trigger was executed. To fix this, we reviewed our schema design and made the necessary changes. For future reference, it would be helpful if the backend functionality was implemented and tested in small phases rather than if the backend was entirely implemented at once.

Kevin: A technical challenge we faced was when displaying data onto the frontend. The output from the backend we received was a JSON which contained an object of objects so we had to use the map function in React(Javascript) as the solution to display our data. At first we were having trouble using this map function as it only works for an array of items; however, our data was an object of objects so we just typecasted that object to an array which allowed us to display our data properly.

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

Starting with ideas, the first thing we had in mind for this was that we wanted it to be a prediction for all NFL games, however, one thing we changed mid-way through was the ability to predict it on the weekly basis and instead based it on the team.

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

For the future, I think that for the application we did not fully get to create an advanced rating system. We wanted to create a more advanced rating by using yards of separation, offensive likelihoods, defensive acumen, coach's playcalling etc. All of these metrics were at our fingertips however they were hard to implement in the time given and this is something we are going to implement in the future by creating advanced queries and working with them extensively to create a unique aspect to the application itself. Also

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

We managed teamwork well, mainly because we communicated well and made sure all
of us were comfortable with the work that they were doing. Most of the first half of the project, we all pitched in ideas and created our database design effectively. Also, something we really emphasized was doing the work together, so there were a few times during the semester when we broke off individually to do work and tried to keep everyone on the same page. That allowed us to share ideas and also for everyone to learn the implementation given the strengths of our teammates. The final division of labor was that half of us were working on creating ideas and implementing them into our advanced queries to see what we

would display within the constraints and requirements of the project and then communicate that to the other half that was working on the frontend to make our application comprehensive and easy to use whilst using implementing the backend all together. However, we all made an effort to try to learn all three parts of the application from the database to the backend, and the front end.