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).

Our final project is almost the same as our original proposal. The only change that we made is that we took out the functionality of the user inputting their own PC data to search for games and replaced it with the functionality to favorite games because we thought it would work better for our ultimate goal of adding a friend system and having it more social media like.

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

We think what our application achieved so far is the basis for what we have planned. Like we said previously, our final goal is to make a social media for video games. So far we added the library of games and users and other things too like a search and the ability to favorite games. We think right now the application is useful as a video game database, but we have bigger plans for it still.

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

We kept the source of the data and the schema consistent throughout the process. Two important changes made were the removal of UserPC and the addition of the likes table in the database.

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?

One thing that we changed from our original ER diagram is that we scrapped the PC table in order to flesh out our Reviews table. We think this is a more suitable design because early on we wanted to create a database where people can discover games through searches, which is why we needed their PC specifications, but we pivoted to wanting to get those recommendations from friends, which is why we started to focus more on the reviews the users were leaving. Secondly the addition of the likes table provided a completely new feature to users where users were allowed to 'like/favorite' a certain game. The likes table would hold the user_id and game_id as foreign keys and a creation time stamp linking both the user and games tables which enables more features and functionalities to be added on top of what we already have.

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

As previously stated, we took out the PC specification search feature and added a favorites section because early on we wanted to create a database where people can discover games through searches, which is why we needed their PC specifications, but we pivoted to wanting to get those recommendations from friends, which is why we started to focus more on the reviews the users were leaving. With the addition of the likes table we could now have a dynamic daily ranking system of games which would display the current most liked games on the website from now and 24 hours prior creating a popularity ranking in which games are in competition to one another. As

games are liked by users the list of the most current liked games are added and those games that have had no activity in the past 24 hours are removed from the list. Such a ranking system effectively portrays user trends and can aid users in decisions related to game purchases. Secondly, the likes table can enable users to access liked games with ease without having to search them up again and just making the experience more convenient.

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

The advanced database programs selected for use in this application were the stored procedure and trigger combination. The trigger allowed the maintenance of the dynamic daily ranking system as mentioned previously. The trigger would activate when a user liked a game (an insert on the likes table) this would then check the table holding the daily ranking: gf_fan_fave table and if the game is already present it would update the creation timestamp and if not it would a new row and all entries older than 24 hours would be removed. The dynamic nature of this feature would not have been possible without a trigger and complements the application by providing a core feature that we want the users to have.

The stored procedure on the other hand combines many complex advanced queries that weight games based on metric such as critic scores, price and they are included in the fan favorites list to make a table that holds all the games that fall within the specified thresholds into a new table which can be easily accessed each time a user refreshes

the page without the need to have the query run multiple time the stored procedure gives access to this data in a very straightforward and efficient manner. Hence it complements our application by improving efficiency.

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.

Rahul - One technical challenge that the team encountered was how large our datasets were. They were causing our apps to load slowly and at some points even took up 100% of the computing power which didn't let anything else load. We ended up writing our SQL queries to combat this, but it may be more efficient to change our database design in the future to make sure we do not run into this problem again.

Nimish - The biggest technical challenge that I had was making a database that was efficient for what we were trying to achieve. The schema must be able to store and retrieve data quickly and accurately, while also accommodating for any changes or updates to the data model. The organization of user data is so important because of how large our datasets are. It would be impossible to search through every game when we make our SQL query so we have to split the data accordingly.

Asif - Incorporating the advanced database programs with the front end presented some challenges as the introduction of a new table had to be properly communicated to other

members and familiarization was required. The subquery in the stored procedure took a while to execute and was later overcome by removing the subquery and joining the results of the previous query with the relation and it improved execution time significantly. Also, the way tables were being accessed were different amongst members which led to the front end and back end not connecting as they should be and that had to be remedied by adjustments to the queries in the stored procedure and trigger.

Navin - A big technical challenge that we encountered during this project was finding an advanced query for our web application to represent. With such a huge amount of data in our database, it was difficult to figure out where to make our down-selects and find something meaningful to produce that can be useful for the end user. We came back to what we had our initial scope of the project look like, and narrowed it down to some ideas that we realized a game application like ours could implement that would make sense to what the user would look for. We also had to make sure to debug and test our query multiple times to make sure it was consistent in the data it outputted and the time it took to load.

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

Apart from what was discussed earlier, there were no other changes from the initial project proposal.

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

One thing that we need to improve our application on is the efficiency of our SQL queries. Right now we have millions of data entries, so parsing through all of them takes up a lot of time. We have to come up with either a more efficient table model or more efficient SQL queries.

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

The final division of labor ended up being fairly equal. Our strategy was to do most of our work during team meetings and then assigning the remaining work amongst ourselves. We left it up to ourselves to keep ourselves responsible and keep up with the rest of the group.