411 Project Reflection Report

1. Please list out changes in directions of your project if the final project is different from your original proposal:

   One of the biggest differences between our original proposal and final project was our creative component. Initially, our creative component consisted of making a graph with the differences in GPA's of different classes for students. This would've helped the user to visualize what classes are best for the user to take. However, we struggled to achieve this goal and thus the final project differs from our original project proposal.

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

   Our application allows users to view data that would be very useful for students who recently joined UIUC such as freshman while being secure. Our database stores usernames, passwords, and unique user ids to maintain security.

   In terms of how our application succeeded, the querries we have included are very useful for students, especially Freshman. We have advanced subquerries that allow users to view the highest GPA 100-level courses, the top courses ranked by average GPA that were offered in the last five years, and a list of the average GPA of each course at UIUC. Each one of these results can prove incredibly useful to students in terms of finding a GPA booster or selecting which gen-ed they should take.

   That being said, we were all new to frontend coding and thus the visual appeal of our application is lacking. The data we display for the result of each query is not formatted well and this can make it difficult for users to digest the data we return.

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

   We did not change the source of our application but we did remove data from it. Our data was the GPA dataset for UIUC from 2010-2021 which had more than 60,000 entries. We changed it to be just 2015-2021 so that our program can run efficiently and show the users what are the GPA's of classes in the most recent years. Furthermore, we added dummy data to some of our tables as we could not access the real netID's and passwords for students.

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 final table implementation has remained more or less the same as what we discussed in our ER diagram. That being said, a more suitable design might be to remove the friends table and just have the Friends w/ relation be from the Users table onto itself. Further, we could store the GPA of each user within the User table. This would provide us with more visibility into users and their GPA along with allowing us access to the GPA of their friends.

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

   One of the functionalities that we added was the length of the password on an update. By using triggers, we made sure that if a user is updating their password the length should be more than 5 letters for better security. One thing we removed was finding the GPA of a particular Professor. We did not think that this made sense as our database used 5 years of data and mostly professors just teach 1-2 classes. Having a specific function to run this would not be too useful for students.

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

   Our advanced database programs completed our application by polishing it off and providing extra functionality. As mentioned above, we used triggers to ensure that when a user changes their password, they have added security so that the new length of their password is above five characters. If not, we set the error text within our trigger to display to the users that their password was not of a sufficient length.

   We also used stored procedures to add functionality to our website by storing our advanced queries in one function so that users could easily click on the button, which would then call the stored procedure, and then view the results of both queries. We did this as our queries were both centered around determining classes with high average GPA, and thus we thought that grouping them together would help our application feel more complete.

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.

a. Lisa: One technical challenge we faced was that the dataset we had had too many entries and thus running queries and especially EXPLAIN ANALYZE on this dataset was extremely taxing and took a very long time. We solved this by reducing the size of our dataset and included the most relevant few years to maintain the usefulness of the data.

b. Sahil: We also faced a big issue when we were trying to code as we stored our database and our server on GCP. Initially, we coded through the GCP SSH terminal but this was difficult as coding on a terminal through nano or vim was hard and unfamiliar to us. When we tried to use SSH from our local machine on VSCode in order to have a better text editor, we ran into many issues as the SSH connection continuously would go offline and costed us a lot of time waiting for it to go back online.

c. Udit: Another issue was that we quickly ran out of credits when hosting our server on GCP. We used up the initial credits that we were granted quite quickly as we had forgotten to turn the server off when we weren't using it. Thus, we had to get more credits from GCP to complete the project and it required us to restart our compute and sql instance, which was a further time waste.

d. Arnav: One of the biggest techincal errors we faced was when we were handling our frontend code. We were using Node.js for our server and styling the webpages with html, but we found it very difficult to access the results of our queries in the javascript code in order to style it. We tried using embedded javascript but the object we were passing was just being recognized as an Object, and the contents within were not being passed. That is why our frontend was not able to style the table we passed it as we had to pass it a jsonified list of the query results which we weren't able to format.

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

   Other than the interface not being as nice and the visualization, we wanted to create a scheduler through which the user can enter the classes they want to take and the application could suggest what is the expected GPA for the student if they take the classes. This could've really help upperclassman when choosing their electives.

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

The application can improve on is making a search bar. Through this search bar, a user can just select a course and get the average GPA of different classes and get the average GPA of all instructors who taught the class. Lastly, we can have a search bar which can be used to lookup a professor or a specific class and give the data of that specific class

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

    Arnav: Created the dummy data for user, friends and reviews table. He also wrote the for basic CRUD functionality for the application.

    Udit: Created the trigger and stored procedure for the application. The trigger made sure that the user password was longer than 5 characters and the stored procedure combined both of the advanced queries together

    Sahil: Team manager for our group. Made sure that everyone was completing their parts whenever it was required. He ran the application on the GCP and created the entire applications interface.

    Lisa: Worked on the advanced subqueries that returned the GPA of courses that could be used by the user. Connected the database to the GCP.

    Our teamed worked really well together. Everyone was communicating properly and asked questions whenever required. Everyone contributed to the project and Sahil made sure that the ship was running smoothly.