Project Report

(Each question is answered on a different page)

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).**

   We had to simplify our application slightly from our original proposal. A few of the features we wanted to include were:

   a) Features to show the correlation between current events and trending videos - this analysis would involve a lot of outside work not relevant to the database aspects of this class.
   b) A video recommendation system that would use the title of an inputted video to suggest other videos a user may like.

   Overall, our final application follows the general sense of what we aimed to create, however, some features were not included due to high complexity and time constraints.

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

   We believe our application achieved its usefulness. Our goal was to create a website for content creators, specifically on Youtube, that would help them improve their titles and content based on popular trends. Our final application provides users with several features that allow them to gain insight into what titles, videos, and keywords are trending, allowing them to generate more views as they produce content of their own. One feature in our application shows the user's most popular videos within our database. A second feature allows users to input a chosen video title and returns a better title that has a chance of garnering more views. We also have a visualization on our website that allows users to see the most popular video based on their category name. All these features help users understand trends in popular videos and use this information to create more widespread content.

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

   We changed the relational schema as we built out our final application. We realized that the organization of our TrendingKeywords table did not make a lot of sense. We added two relational tables KeywordVideoMap and KeywordCategoryMap which allowed the database to be connected more coherently. We did not change the source of the data for our application. We maintained the same dataset of Youtube Trending Data from this source: https://www.kaggle.com/datasets/rsrishav/youtube-trending-video-dataset?select=US_youtube_trending_data.csv

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?**

   As mentioned in the point above, we changed our relational schema to allow the relational connections between tables to make more sense, thus, becoming more usable in the development of our application. The two new relational tables mentioned above would create a connection between TrendingKeywords with VideoInfo as well as between TrendingKeywords and CategoryInfo. This was a more suitable design because it allowed us to perform more efficient queries on the data in a SQL style. Without these additions, more of our functionality would have relied on complex backend code rather than database queries.

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

   When we first planned out our webpage, we had a large number of functionalities that we wanted to incorporate. Due to time constraints, we had to remove some of these functionalities. Initially, we wanted to create a video recommendation system that would use its users' searched titles and categories to suggest other videos for them to watch. This functionality is tricky to implement without machine learning algorithms or natural language processing, hence we weren't able to do so within our time limit. We also wanted to analyze website traffic in order to generate times at which video exposure would be highest, and also compare video publish times and dates based on current events around the world, however, we were unable to complete these as well

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

   Our first advanced query retrieves the video titles and the view counts for the top 10 most viewed videos that were uploaded by creators who have one or more videos with over 1 million views. This query complements our application as it allows our users to see which keywords would best aid in video exposure. By filtering our query to only include creators with one or more videos over 1 million views, we allow the user to gain insight into what titles have been generally successful for other large creators, rather than being shown results for a one-time viral video. Our second advanced query counts the number of videos for each category which has a title that contains the most commonly used keyword. This also complements our application as it allows the user to see how many videos are using popular keywords for each category. Based on this data, they decide on the effectiveness of each keyword, and can then decide whether or not they would like to include it in their video title for maximum video exposure. Our stored procedure allowed the user to choose filters on the dataset between a specific date range to create a visualization as well as a sortable table. This allows the user an intuitive experience to transform a large amount of data into a simple graph where they can easily determine the parts most relevant to them.

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.**
    - Sritha: One challenge we had while developing our frontend was creating our data visualization. The issue we had was that our data was in a CSV file and we had to reduce it to two columns but we could not manually go through the document to do so. To fix this issue, we created a function using javascript that would take our CSV file as input and return only the needed columns for our visualization.
    - Ayush: I had issues deploying our app to GCP. I am not exactly certain where things went wrong. Since we did not develop in a virtual containerized environment, perhaps some of the dependencies did not get passed correctly when deploying to GCP causing issues. I think this can be fixed if the end goal of deploying a full-stack app to GCP is maintained throughout development. Using proper virtual environment policies to maintain dependencies could fix this issue, but our group was unable to overcome this challenge.
    - Yue: I have a problem while finishing the front end. I want to apply the same style module to all pages. I wrote a CSS file, but not all HTML pages are suitable, they need separate CSS files. The background of some pages is not displayed. I'm not very skilled and don't have enough time to do the front end of all the pages. I also had trouble adjusting the position of the image.
    - Riya: I ran into issues while designing some of the frontend pages, specifically with styling for CSS and HTML. I was unable to make certain changes - including creating specific buttons - using HTML. This is because HTML syntax is not able to employ CSS style sheets. Another more general technical issue I ran into was not being able to clone the repository from Github or push my changes directly from visual studios.

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

   In the original proposal, we planned to give users a visual representation of how their content was performing compared to other videos in their category and desired audience demographics. However, audience demographics are not shown in our final application. Another change compared to the original proposal is that we did not implement the feature allowing users to track their own channels and monitor the impact of their content over time.

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

   I think we would want to implement the features that were left unimplemented from our original proposal. These would unify our application into something unique and cool for users to use. Given additional time and resources, I think these features would be doable to implement. We could also think about how to partition parts of our database to improve the efficiency of queries since our dataset is somewhat large, with ~40K entries. Furthermore, I think we would want to improve the security aspects of our application and then deploy it on a live server. Since we were mainly focusing on the end product, we did not put a lot of consideration into security flaws. If we confirm that our application is safe from cyber threats, then we could deploy it for others to use which would be amazing.

10. **Describe the final division of labor and how well you managed teamwork.**
    - Ayush: Worked on backend for CRUD, stored procedure, and trigger. Worked on frontend for stored procedure. Worked on Flask endpoints to connect backend with frontend. Worked on writeup report.
    - Riya: Wrote one of the advanced queries. Worked on finding ways to index for stage 3. Worked on finishing up the form page, including fixing and styling all its linked pages. Helped to debug issues on both front and back end. Helped research Flask for front and back end connection.
    - Yue: Worked on frontend for page appearance.
    - Sritha: Worked on developing frontend interface. Created the page layout and added a navigation bar. Developed visualization on the website using HTML and D3. Used JavaScript to create a function that would parse CSV files to help create a visualization. Worked on writeup report.
    - For all 5 stages, we met over zoom/discord as a team. Someone would share their screen and we would collaborate to finish each stage. That meant helping debug code as we ran into issues, researching how to implement certain features of our project, or simply meeting up to discuss how to split up work for the upcoming week.