Acme would like to improve the security of their database logon and requires a suitable method which will prevent common attacks on the database. Therefore, they request a password system for administrators and Acme personnel. Create the appropriate users and groups in the movie database so administrators can log on securely using a password (or suitable biometric alternative). The password must pass a minimum complexity test. This can be achieved at the Database level using users and groups. Ensure you have a full backup before doing this task, Keep you UserNames and Passwords simple.

Create a query that will calculate the top 10 most popular movies each time the Top 10 web page is opened. The query should also update when a user adds a rating to a movie. The data from the top 10 movies should be appended to a new streaming table in the movie database each time there is a change in the top 10 ratings. Using the data from the streaming table to create a chart front end to display the data. The front end should be refreshing automatically to show the changing values in the data. You may be required to use an external API or plug in. Your final solution may vary depending on the database implementation (local server, cloud, etc), contact your CITE representative for clarification. There are several points:

1. Create a sperate linked table in the Movie DB that will hold the values for each movie when it is the result of a movie search. This value is incremented after each search. The Movie DB and the table (streaming table) is constantly updating as users search for movies; think of thousands of users each hour (ie IMDB). The increment is a simple update query. This data can be presented as live stream of data and will change over time. You will simulate this feature. The real world solution would be hosting the table and DB on Azure or AWS, this may not be possible, but the whole process can be seen as “analytics” similar to collecting data from a temperature sensor and then displaying as a graph over time. Historical can be taken as the table grows. Your final solution can encapsulate a range of options; it is semi-vague to allow your team the scope to implement different solutions.
2. Append a new column(s) to the movies table which can store a ratings value (if this is not already the case). This value can be edited each time the movie is rated by a user. Therefore, the data can be presented as top ten five star movies.

Implement code and performance optimisers on all appropriate code and document this process. Create a suitable **Optimisation Report** for inclusion with your **Software Development Master Document**. As the question suggests, how will you ensure your final solution on the Server is running as fast as possible, can it handle the high volume of traffic? Consider these issues before you hand over the final project. You will need to ensure your code is fully optimised for speed. Consider latency, and fetch times when a query is submitted to the DB via your HTML frontend and PHP backend. This is a report on possible options that is forwarded to the client for their consideration.

Contact the CITE representative and discuss the all requirements to ensure quality of development and final application (minute this meeting for inclusion into the Software Development Master Document).