**C++ TASK 2:**

Task: Implementing a Distributed File Sharing System with Poco C++ Libraries

Summary:

In this task, you will design and implement a distributed file sharing system in C++ using Poco C++ libraries, with Yugabyte serving as the storage backend. The goal is to create a secure and efficient system that allows users to upload, download, and share files seamlessly.

Day 1: Project Setup and System Design

Summary:

On the first day, you will set up the project and lay the foundation for the distributed file sharing system. You will configure the development environment, install Poco C++ libraries, and outline the system's architecture.

Tasks:

* Set up a new C++ project for the distributed file sharing system using Poco C++ libraries.
* Install and configure Poco C++ libraries for networking and database access.
* Define the system's requirements, including user authentication, file storage, and sharing mechanisms.
* Design the overall architecture of the file sharing system, including components and interactions.

Day 2: User Authentication and Registration

Summary:

The second day focuses on implementing user authentication and registration features, ensuring that users can securely access the file sharing system.

Tasks:

* Implement user registration functionality in C++, leveraging Poco C++ libraries for user management.
* Develop user login and authentication mechanisms to verify user identities.
* Ensure secure password storage and handling.
* Test user registration and login processes with sample user accounts.

Day 3: File Upload and Storage

Summary:

Day three is dedicated to enabling file uploads and secure storage. You will implement features that allow users to upload files to the system.

Tasks:

* Implement file upload functionality in C++ using Poco C++ libraries for handling file transfers.
* Design a storage mechanism to store uploaded files securely in Yugabyte.
* Ensure proper error handling during file uploads.
* Test file uploads with various file types and sizes.

Day 4: File Download and Sharing

Summary:

On the fourth day, you will implement file download and sharing capabilities. Users should be able to access and share files within the system.

Tasks:

* Implement file download functionality in C++, allowing users to retrieve files securely.
* Develop sharing mechanisms, such as generating shareable links or specifying user access permissions.
* Test file download and sharing features with different user scenarios.

Day 5: Testing, Documentation, and Optimization

Summary:

The final day involves comprehensive testing of the distributed file sharing system, documentation, and optimization efforts.

Tasks:

* Conduct extensive testing of the system, including user authentication, file upload, download, and sharing functionalities.
* Document the setup instructions, usage guidelines, and key implementation details.
* Optimize the system's performance, considering factors such as file storage efficiency and response times.
* Prepare a brief presentation or document highlighting key design decisions, challenges faced, and optimization strategies.

End Summary:

By the end of this task, you will have successfully designed and implemented a distributed file sharing system in C++ using Poco C++ libraries and Yugabyte as the storage backend. The system allows users to securely upload, download, and share files, with robust user authentication and efficient file storage and retrieval mechanisms.