CSEN241 - Cloud Computing **Project Proposal**

Project Title: *Flow Disk — Cloud Storage Application*

Team members

Name	Email
Zichao Wu	zwu2@scu.edu
Chi-Hsien Wu	cwu7@scu.edu
Yihui Qin	yqin2@scu.edu
Tengjiao Zhu	tzhu2@scu.edu

Project Description

Main Goal:

This project aims to practice cloud file storage techniques using multiple servers, cloud storage, and distributed systems. It is safe to say that a user’s data is stored somewhere in the cloud and it is safe to say that it can also be accessed by people and get changed. We will attempt to create a secure and robust file system so that there will always be a backup to cloud data if some is lost in unexpected interactions. We want to also optimize the data transfer traffic by distributing files in balanced chunks and using hashing techniques to track each data block. If time allows, we want to make the project a service that is available on the internet (kind of like a SaaS).

Motivation:

Developing a cloud-based file sharing and data storage application is a captivating journey driven by the increasing need for efficient and secure digital file management. This project aims to provide users with convenient access, boosting productivity and collaboration. The exploration of cutting-edge technologies and software development aspects contributes to personal and professional growth. The potential positive impact on users' lives, coupled with monetization opportunities and an expanded product portfolio, serves as strong motivation. Additionally, the pursuit of maintaining industry leadership, community building, and creating a lasting legacy further enriches the compelling reasons for undertaking this rewarding venture.

Cloud Technologies to be used:

1. AWS EC2
    a. Used for hosting our application in the cloud
2. Docker Container
    a. Used for virtualizing our file storage servers
3. Kubernetes
    a. Used for configuring our containers together

Architecture:

The cloud file storage application has two components, the server program and client program. The clients implement the upload and download utility.

The upload utility:

1. Files read are divided into blocks
2. Each block is assigned a hash id
3. Client uploads each block to one or multiple file servers. After successful upload, the client updates the metadata of each file to be filename -> list of (hash id)
4. After the client concludes the upload, it waits for next file to be uploaded

The download utility:

1. The client fetches the list of hash ids belonging to the file from metadata server
2. For each hash id, the client downloads from closest available server
3. The client reconstitutes the original file by concatenating the binary block data

The server program runs on multiple machines, each one of them accepting incoming upload and download requests, servicing the clients. Each file server maintains an in-memory mapping from block id -> block data

The metadata server maintains a mapping from file name -> list of (block id)

The file storage program is dockerized - that is, it is built & deployed as a docker image

Client — get list of blocks of file –> metadata server
Server <— list of block id – metadata server

Division of the task:

1. Proposal Design, and Architecture (All)
2. Backend upload & download client, and server program (Zichao Wu, Tengjiao Zhu)
3. Backend middleware and API (Tengjiao Zhu)
4. Backend DB setup (Chi-Hsien Wu)
5. Frontend file storage architecture and program (Yihui Qin, Zichao Wu)
6. Frontend UI/UX design (All)
7. AWS EC2 setup/configuration (Yihui Qin, Chi-Hsien Wu)
8. Docker containerization (Zichao Wu, Yihui Qin)
9. Kubernetes Orchestration (Chi-Hsien Wu, Tengjiao Zhu)
10. QA Testing(Tengjiao Zhu)