

CSE 6324 - Advanced Topics in Software Engineering

Project Proposal

Simplified Cloud-Based Storage Application in Java

Team Members:

- **Dishant Koli** (1002067615)
- **Niral Shah** (1002061390)
- **Harshada Padwal** (1002080814)
- **Mansi Patel** (1001874295)

Introduction:

This application implements Java cloud-based storage application. Like with Dropbox and Google Drive, the application will put an emphasis on core features and concurrent programming. The user interface may be a web interface, standalone GUI, or command-line interface. The application will support a single user who can connect to the server using various devices (clients). This project's goals are to learn and use concurrent programming techniques, as well as to create a scalable cloud storage system.

Major Use Cases:

1. File Upload:

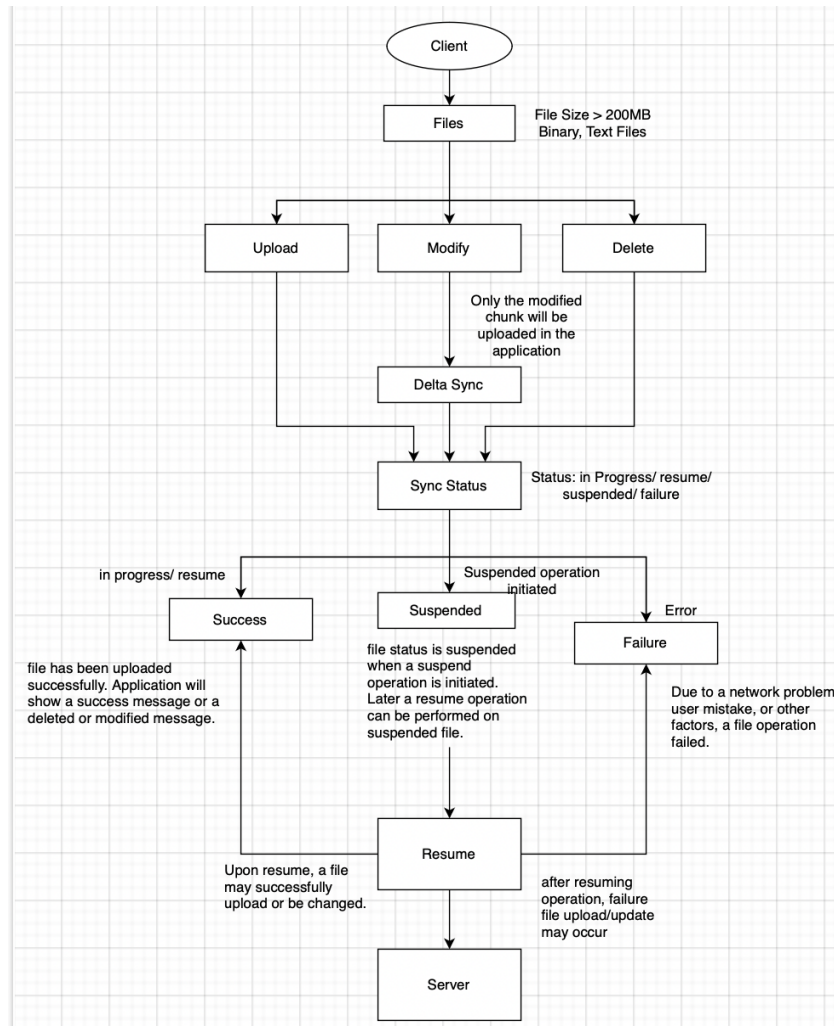
- The client can upload the file for syncing. The client should be able to upload huge files (200MB and greater in size) as well as text and binary files.
- The program will automatically sync any modifications made to the files over time.
- File transmission will be done using UDP.
- If the upload is not halted by a client request or a network issue, the file should upload successfully.
- Block-wise file division and transmission will be used.
- It is necessary to display the file upload status, which indicates whether the files hosted by the clients are in sync or out of sync.
- The file's status needs to be maintained in case a network failure or client intervention causes an upload error, stopping the process for whatever reason. If this happens, the block transfer will continue from the block where the resume was sent. Via automated delta sync, all files on the clients and the server are in sync.

2. File Deletion:

- A file should be deleted from the server when a client requests it.
- If the deletion is not halted by client action or a network issue, the file should be successfully erased.

3. File modify:

- Any changes made to a file at the client end should also be reflected in the file that has been uploaded to the server.
- Instead of changing the entire file on the server, just the portions that have recently undergone client changes should be altered, and each client will receive the updated data from the server to retain the most recent copy of the file.
- It is necessary to display the file upload status, which indicates whether the files hosted by the clients are in sync or out of sync.
- An error message should be shown if an error arises while updating or altering a file that is already existent in the program. Error handling is done to ensure a smooth transfer of the file's content.



Collaboration Plan:

- We will assign tasks to the team members according to their areas of expertise to construct this application.
- Dishant and Nirali will focus on the client-side development, which includes creating the user interface, putting file sync and delta sync into practice, and managing the file blocks.
- Harshada and Mansi will focus on the server-side development, which includes maintaining the file system, putting the file transfer protocol into practice, and handling network connections.
- Our version control system will be GitHub, and we will make distinct branches for each feature.
- Each member of the team will provide assistance to every other team member with their objective.

Team Member Contributions:

- **Dishant Koli**
 - Client-side development, user interface design, file sync implementation.
- **Niral Shah:**
 - Client-side development, delta sync implementation, network protocol implementation.
- **Harshada Padwal**
 - Server-side development, file system management, network connection management.
- **Mansi Patel:**
 - Server-side development, file transfer protocol implementation, testing and debugging.