**WORKING OF THE SERVER**

1. **signup endpoint**

Gets the signup data and creates a user record on the “users” table.

1. **Login endpoint**

Checks whether the given email od has an entry. If exist then checks for the hash(password) matches and returns appropriate response.

1. **Upload**

Checks for the type of provider.

If ‘aws’ then uploads the file to the S3 and updates the record to the database.

If ‘azure’ then uploads the file to the blob storage and updates the record to the database.

1. **Update**

Here, the update can happen in two ways with each having another two possibilities.

1. Updated by the owner, to AWS

Here since it’s a owner access we don’t need timestamps to check the access , simply store the file to the S3 and update to the database

1. Updated by the owner, to AZURE

Similar to the above, the file is stored to the blob storage and updated to the database.

1. Updated by a person who has access, to AWS

Checks for the valid access by comparing the current timestamp received from the users device and comparing to the access set by the owner of the file. If access permitted the file will be updated or else appropriate messages are sent.

1. Updated by a person who has access, to AZURE

Similar to the above, the access is verified by the timestamp and if permitted to access than the file is updated or else appropriate messages are sent.

1. **Delete**

Since the person receiving the access doesn’t have access to delete. The only person to have access with the delete feature is the owner of the file.

First all the shares provided by the owner are scanned and removed from the databases.

Then depending upon the storage provider, the file deletion process is carried out.

After deleting the file the data entry from the database is removed.

1. **View**

It is a display function for the dashboard. Viewing all the uploaded files by the user.

It gets the user\_id and finds all the files uploaded by the user and sends the response as json data.

1. **Download**

Here the download can be accessed by two people either the owner or the person who has access to the file.

So, if the “owner” of the file is accessing, then the storage provider where the file is stored is checked and then the appropriate download sequence is carried out based on aws or azure.

If the access is a “shared” access, then the current timestamp is validated with the timestamp set by the owner of the file for access. If the validation is successful the file will be downloaded or the appropriate error message will be sent.

1. **Share**

The share is a function which can be used by the owner of the file to provide access to the other users.

Hence, the request is received and the access for the share is registered in the database.

1. **sharedList**

this endpoint is used to retrieve the files shared or files received from other persons depending on the filterExpression provided to the endpoint.

1. **revoke**

This function is available on the file owner dashboard, based on the id received the appropriate access record is removed from the database.