Substrate Assignment

DropBox Clone on Substrate

OVERVIEW

The idea is to build a dropbox clone on Substrate as a blockchain with decentralized storage using IPFS protocol. Which will have features to upload/download files. Keep a track of the owners of the files.

SPECIFICATIONS

1. Upload/Download files
2. The files will be uploaded on the ipfs using any of the providers such as nft_storage, pinata, etc. Each file will have a content_id (which is a hash value as a string) once uploaded.
3. The defaut free file size is 250mb
4. If the file size is greater than the default size the users will be charged for every extra MB. (applicable to both upload/download) functions.
5. The charge for excess file size can be decided on your own.
6. The money charged will be transferred to Default Account Dave (So Dave will sort of be the Accountant for our app).
7. File Type is a Enum with variants Normal, Privileged
8. Privileged files can be downloaded by paying for the files.
9. For Privileged files the user uploading will pay extra fees if exceeds the default limit. However, when a user is downloading only the cost of the file will be charged and not for the excess limit.
10. Store the details of the users who downloaded. Also, enabling the number of downloads.

Features

1. Upload File

a. Application will need to store all the details based on the inputs

b. Owner of the file is the account id of the uploader.

c. Users will need to only pass the link of the file.

d. Must take inputs:

   i. file_cid(content_id)

   ii. file_link

   iii. allow_download: bool

   iv. File_type:

   v. cost (only for Privileged files)

   vi. file_size:

2. Download File

a. Update the storages accordingly as per the extrinsic completion

b. Only files that have allow_download as true can be downloaded

c. If file is privileged user will need to pay the cost for that file and won't need to pay for excess file size if exists

d. Inputs:

   i. user_accountid

   ii. file_cid

3. Transfer Owner

a. Can only be called by the file owner

b. Inputs:

   i. file_cid

   ii. new_owner_account_id

Expected:

1. **Storage that stores file information based on users**
2. **Number of downloads per file**
3. **Ensure statements where ever necessary**
4. **Test Cases (Bonus)**

Note:

1. **Version 1.0 does not need ipfs implemented, you can pass just a random id for the file_cid, and a dummy link for the file which is again a string.**
2. **Version 2.0 of the app will implement IPFS storage, Skeletal UI for the app and integration to the blockchain.**