Decentralized Secure Cloud Storage Using Blockchain

Cloud services has gain lots of popularity because of low cost storage and heavy computation servers are available in cheaper cost but this advantage raise a security issue of data security as data will be away from user hand and store at cloud server and cloud server employees or management may misuse and there is no direct way available to know about data leak. Other disadvantage is cloud is managed by centralized servers (single point server) and if this server down then services will not be available.

To overcome from above issues we are employing Blockchain service in cloud, as Blockchain has inbuilt support for decentralized storage where data will be stored at multiple nodes and if one node down then services will be access from other working nodes. Blockchain store data in blocks as transaction and associate each block with unique hashcode and this hashcode will get verified before storing any new block and if any node block get modified then it will result into hashcode mismatch and data leak or tamper will get detected.

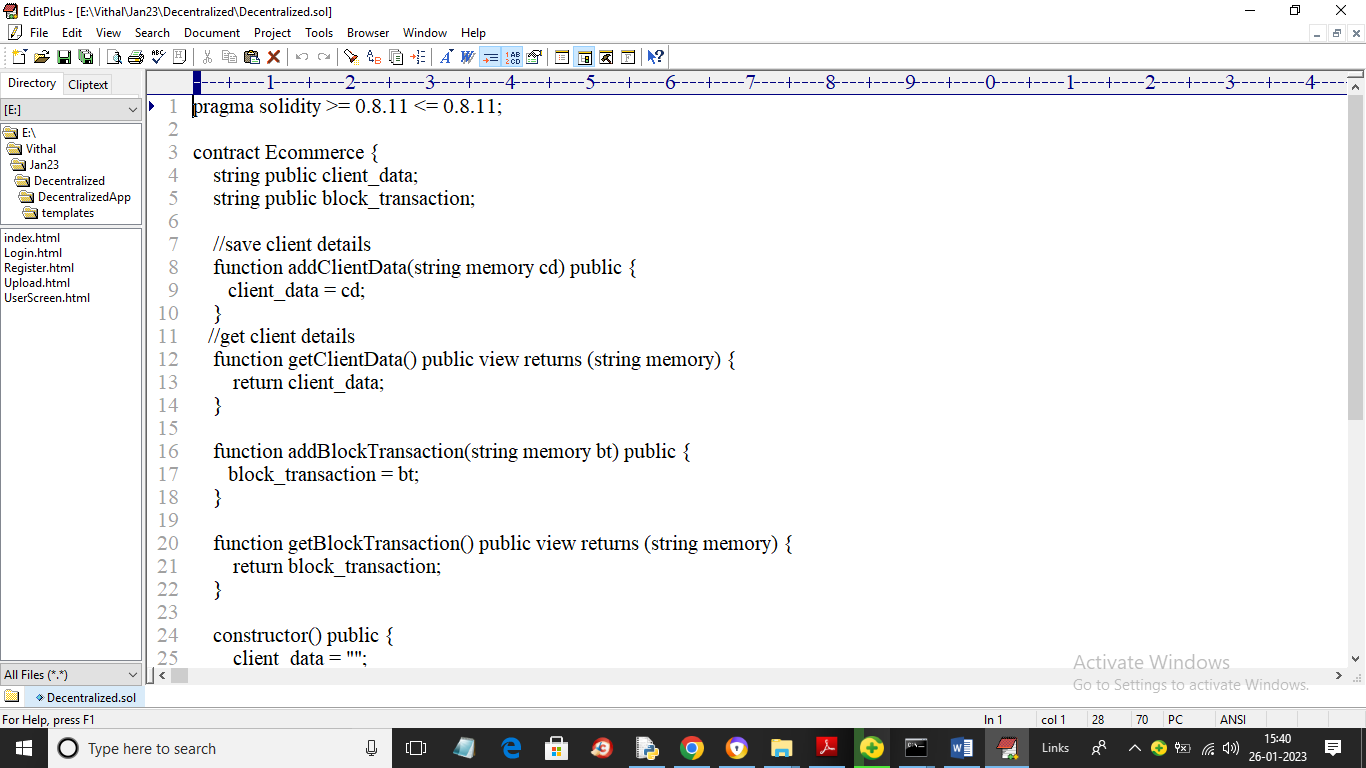
Here we are employing IPFS server to store file data which will return storage address as hashcode and this hashcode will get saved in Blockchain. While retrieving we can get hashcode from Blockchain and then query to IPFS with that hashcode to get file back.

Here if IPFS hack then entire file system will be exposed to attacker so extension we are splitting entire file into multiple blocks and each block can be stored in different IPFS servers and of one IPFS hack then attacker may get some blocks of file which he cannot decrypt or view.

**Extension Concept**: In this project as extension we are using Blockchain and Cloud services to store user data securely. Extension work will divide user file into Blocks and then encrypt each block using AES algorithm and then store each block in IPFS server at different nodes. IPFS will return memory address of stored block and this address will be saved in Blockchain. While downloading application will collect all block addresses from Blockchain and then send request to IPFS with those Block addresses to get all files block and then all blocks will get merge and can be decrypted and download by the client.

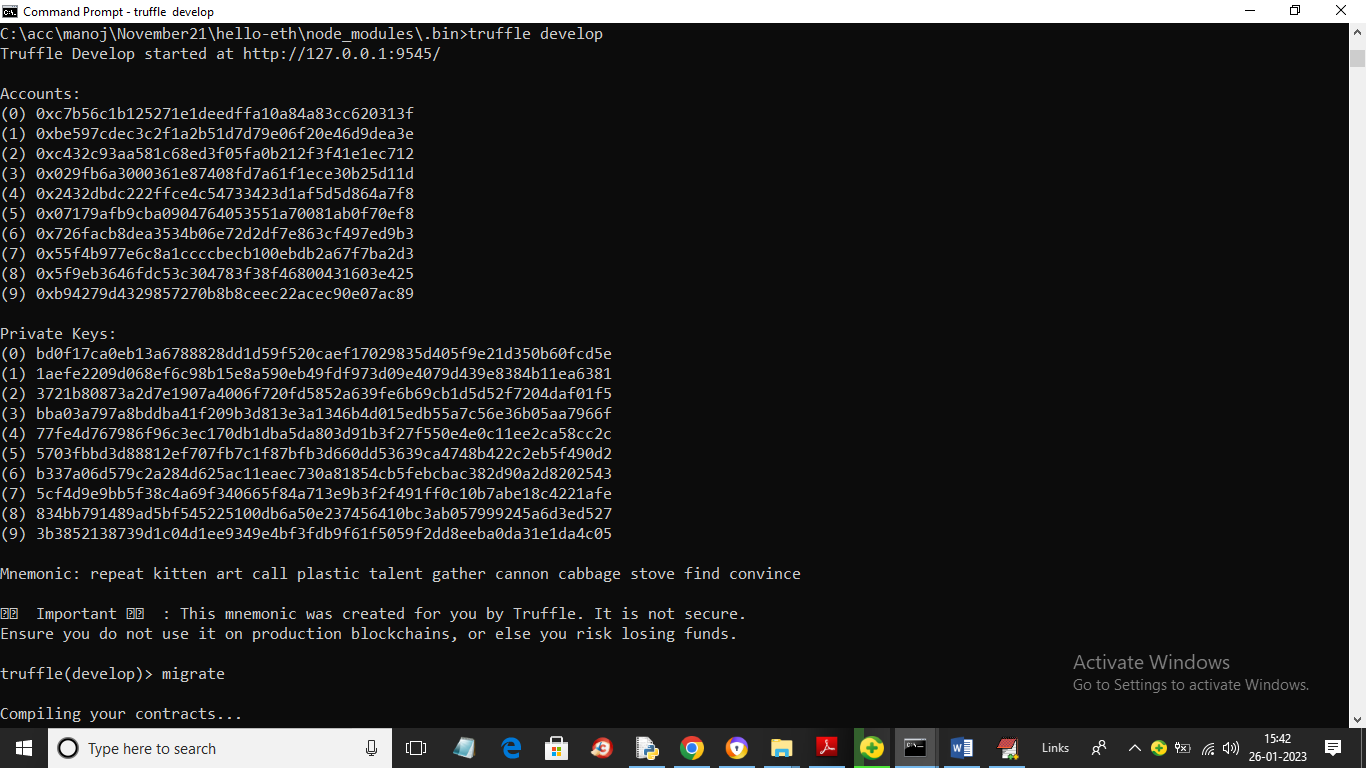
By storing data in different IPFS we cannot allow attackers to know the address of all file blocks and he cannot download the file. Blockchain will store data in immutable format which cannot be tamper so full security will be provided to user data.

To store data in Blockchain we need to design SMART CONTRACT using SOLIDITY code which contains functions to store and retrieve data and below screen showing Smart Contract designed for decentralized storage.

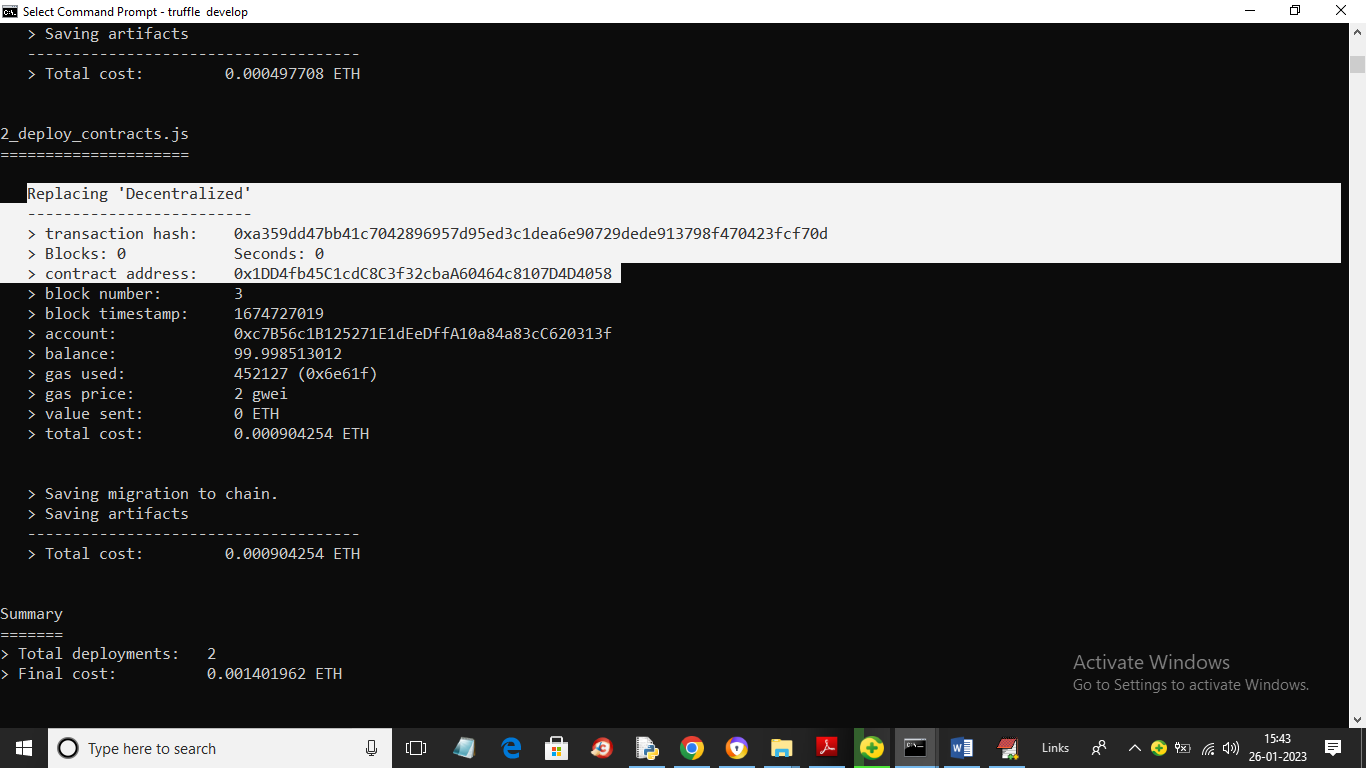


In above smart contract we have functions to save and get client data and file transaction details. We need to deploy this smart contract in Blockchain using TRUFFLE tool by using below steps

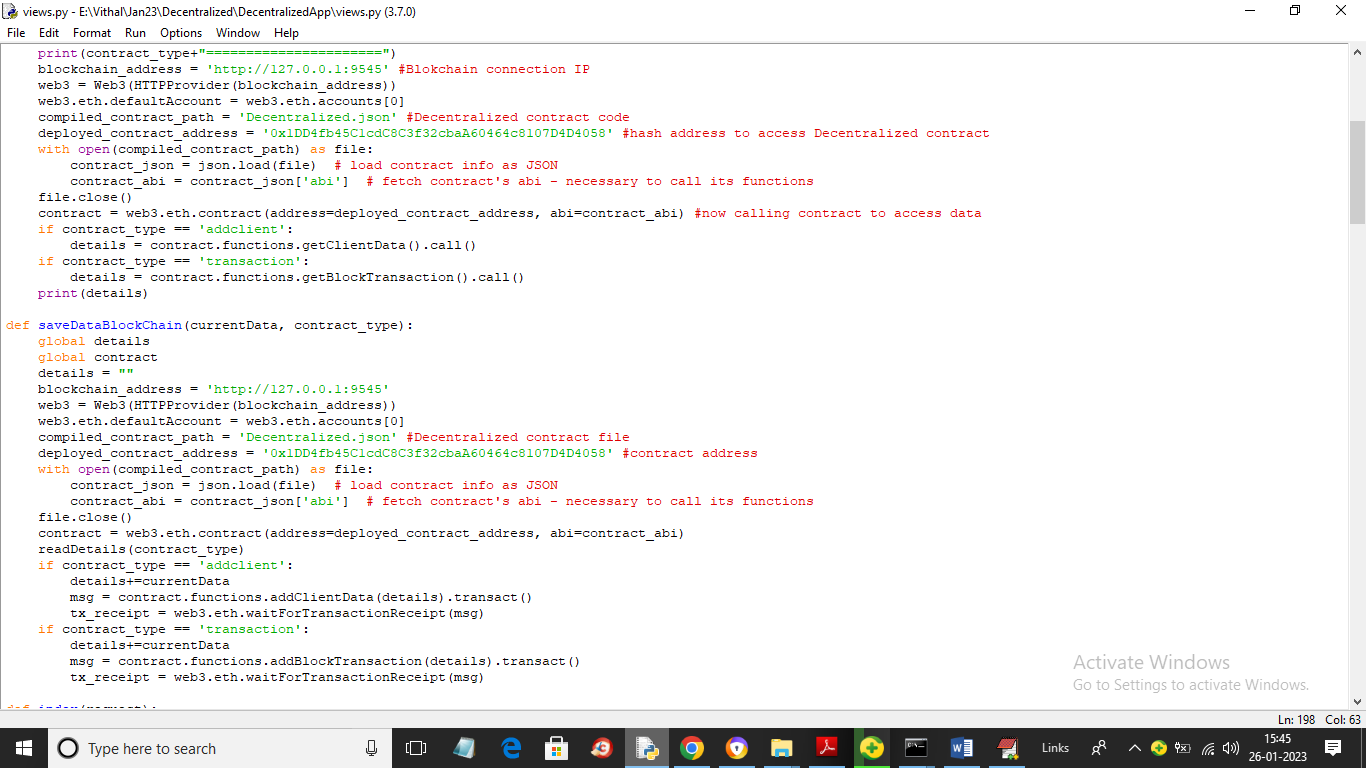
1. First go inside ‘hello-eth/node\_modules/.bin’ folder and then double click on ‘runBlockchain.bat’ file to get below screen



In above screen Blockchain server started with default accounts and private keys and now type command as ‘migrate’ and press enter key to deploy contract in Blockchain and get below output

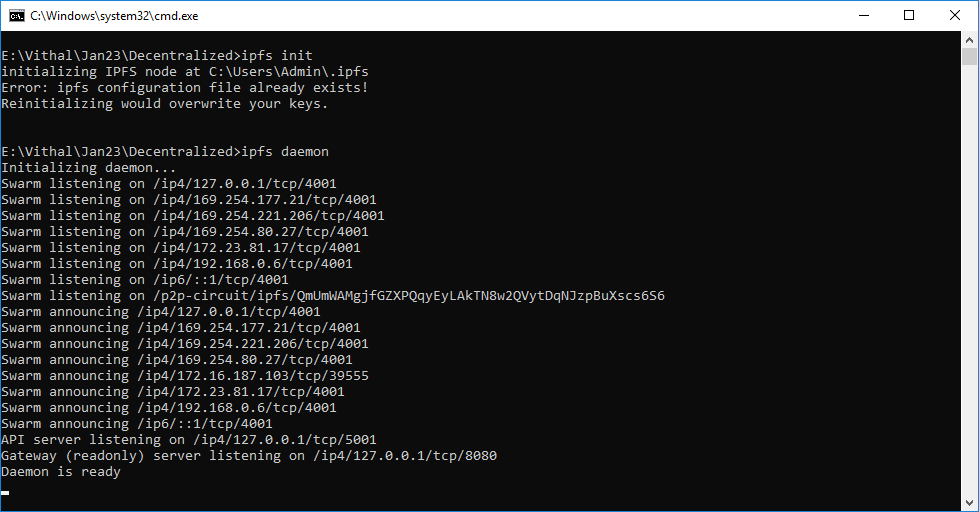


In above screen in white colour text we can see ‘Decentralized’ contract deployed and we got contract address also and this address we need to deploy in PYTHON program to call Blockchain functions to store and retrieve data. Below screen showing python code calling smart contract

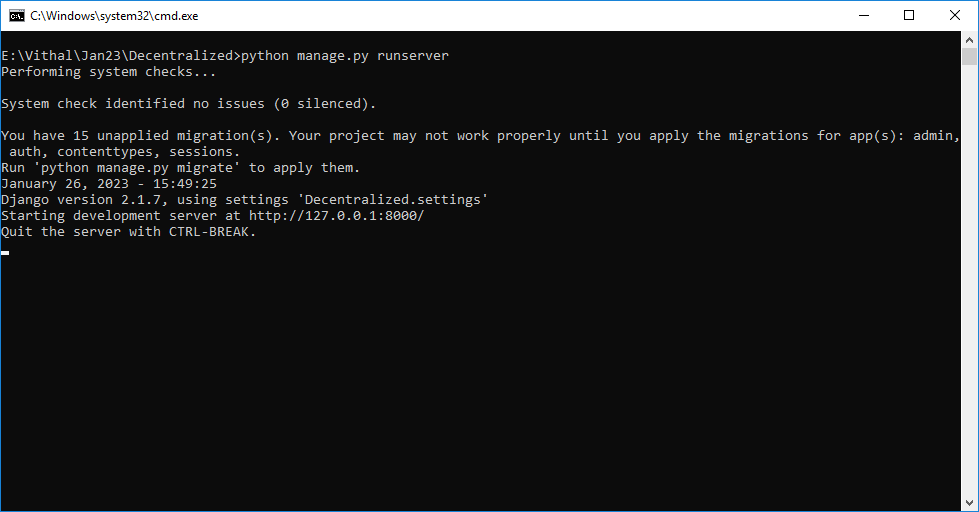


In above screen read red colour comments to know how to call Blockchain functions using python code.

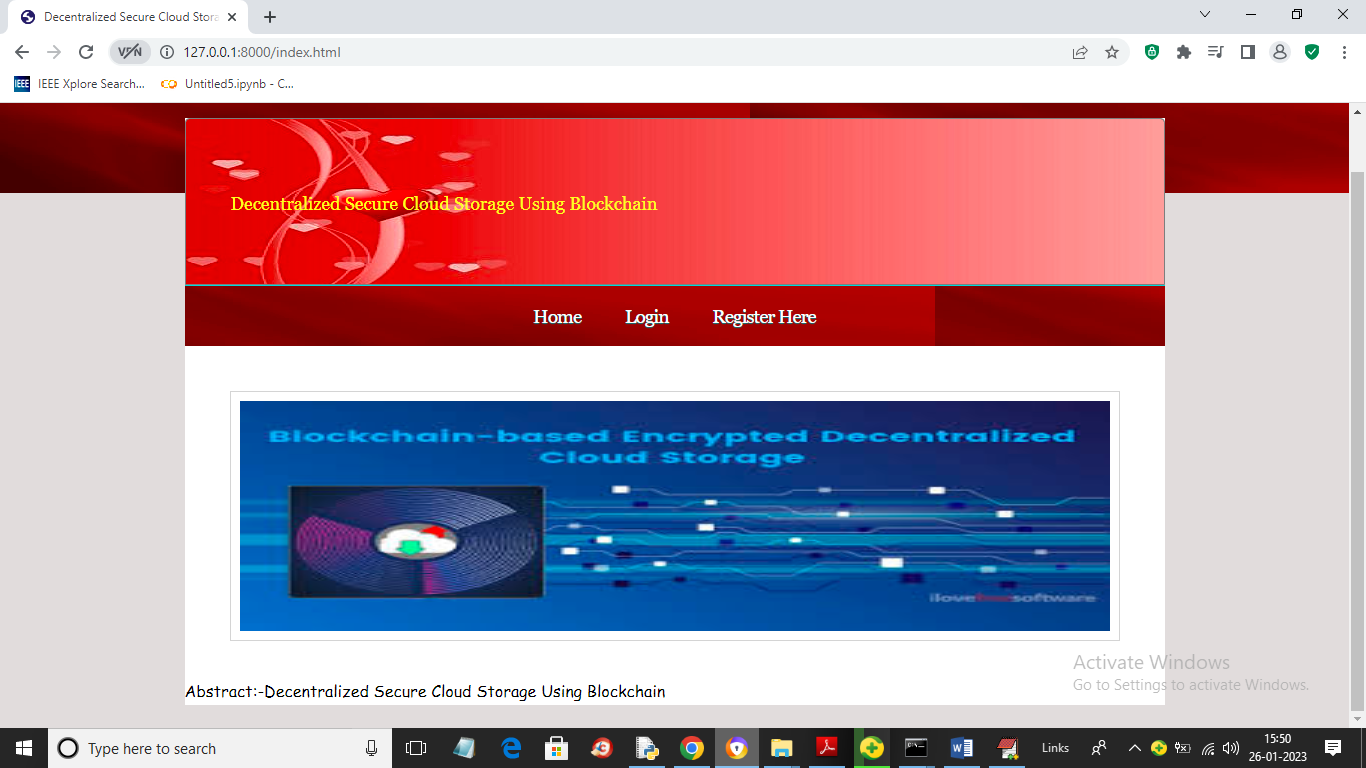
Now double click on ‘Start\_IPFS.bat’ file to start IPFS server and get below page



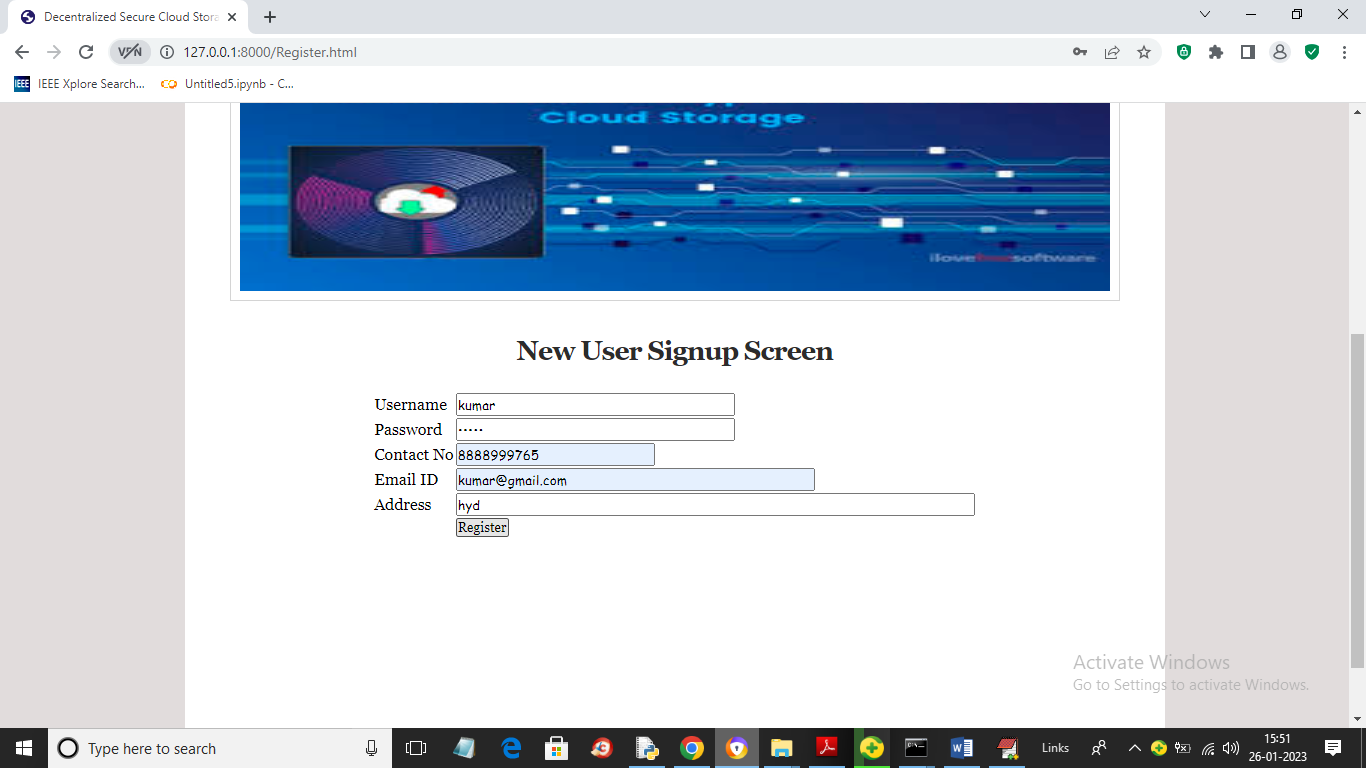
In above screen IPFS server started and now double click on ‘run.bat’ file to start python WEB SERVER and get below page



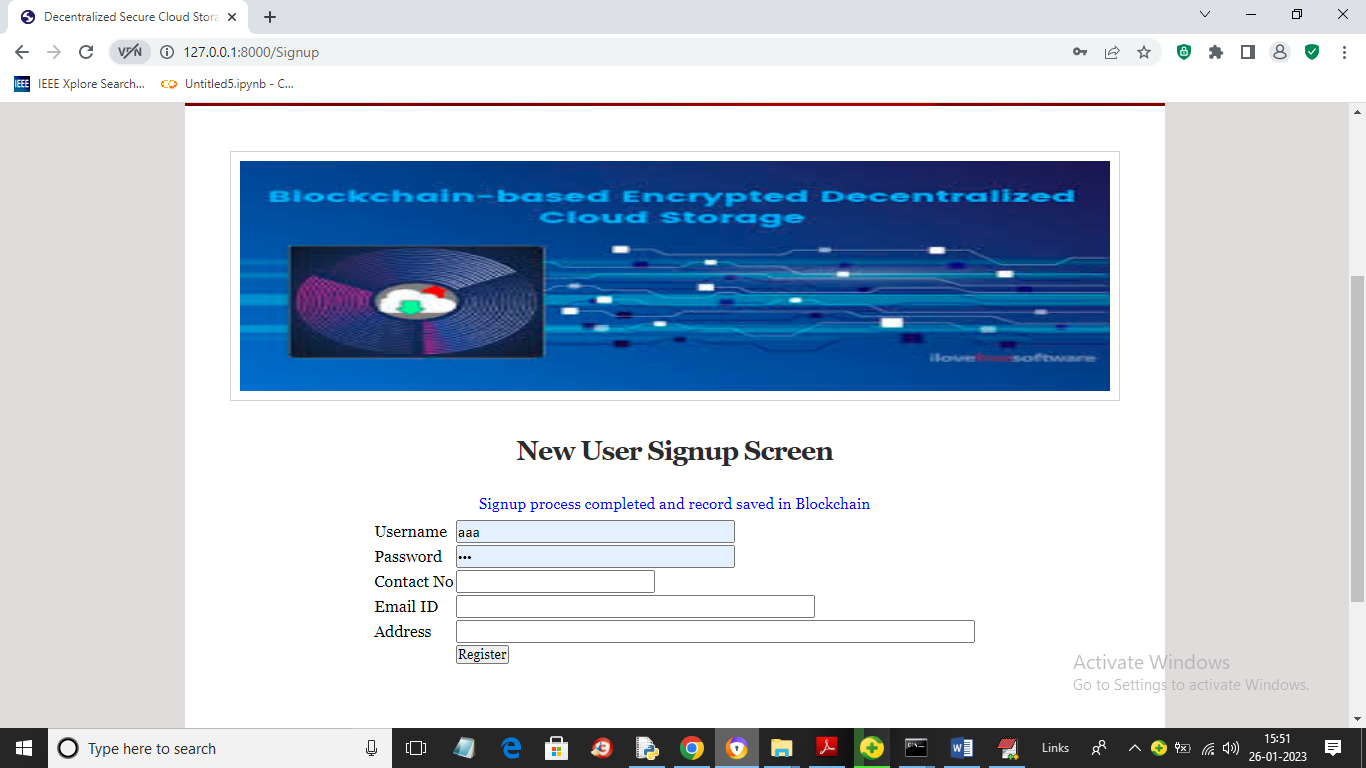
In above screen python server started and now open browser and enter URL as <http://127.0.0.1:8000/index.html> and press enter key to get below page



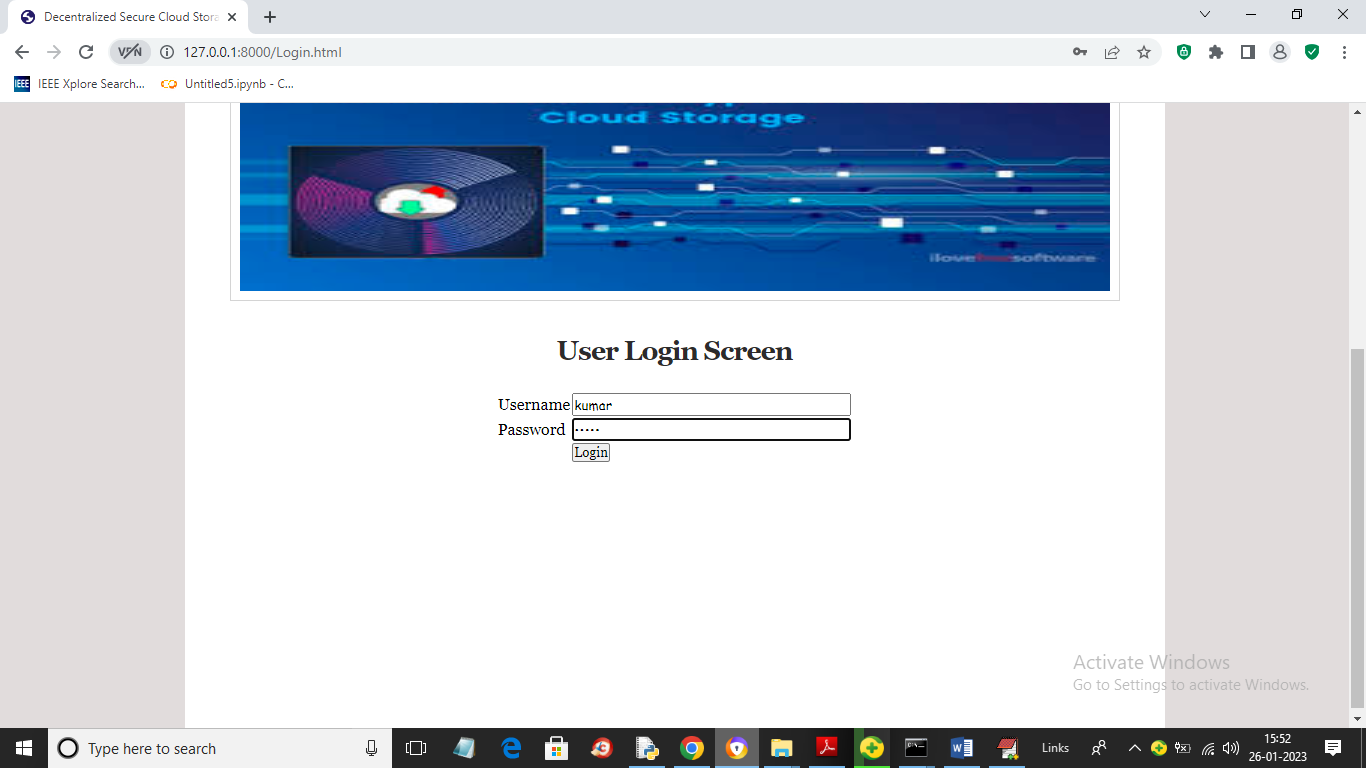
In above screen user can click on ‘Register Here’ link to get below signup page



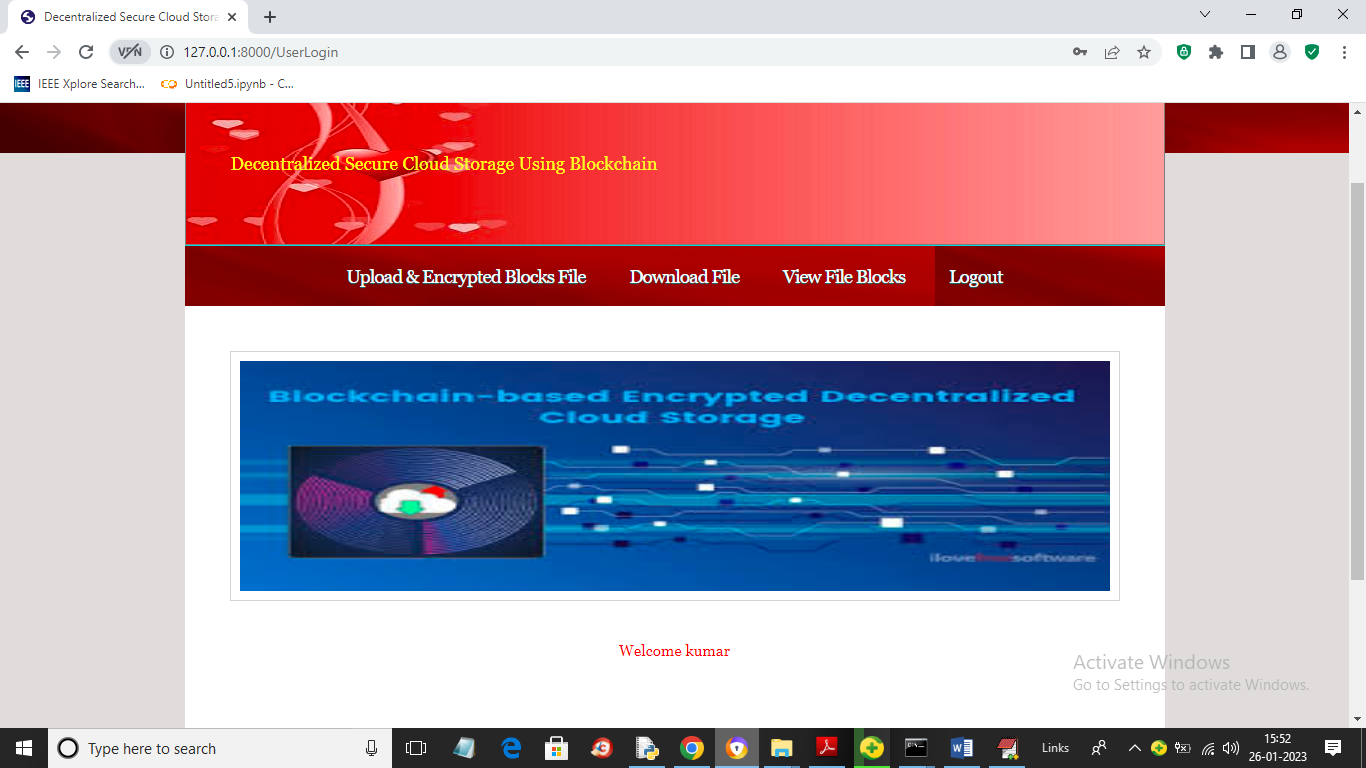
In above screen user is entering signup details and then press button to save client user data in Blockchain and get below output



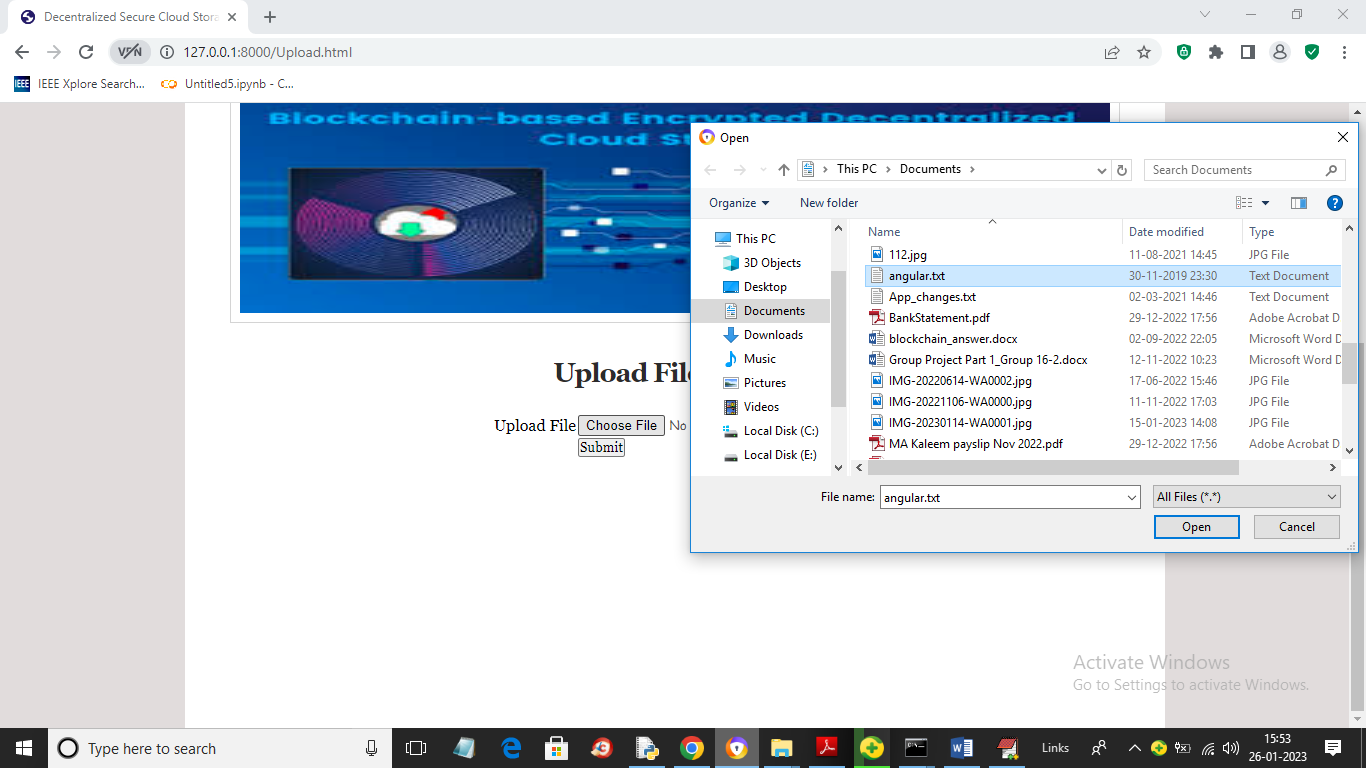
In above screen in blue colour text we can see signup task completed and now click on ‘Login’ link to get below screen



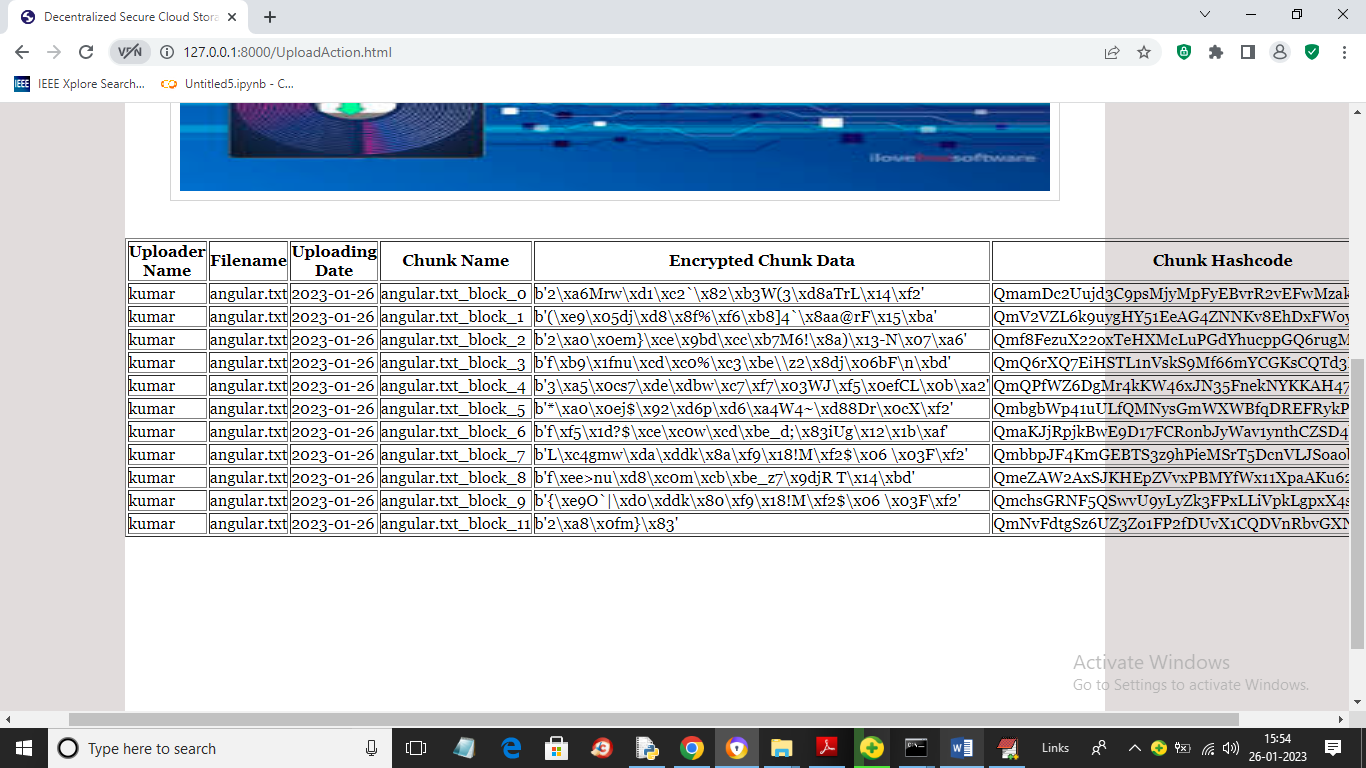
In above screen user is login and after login will get below page



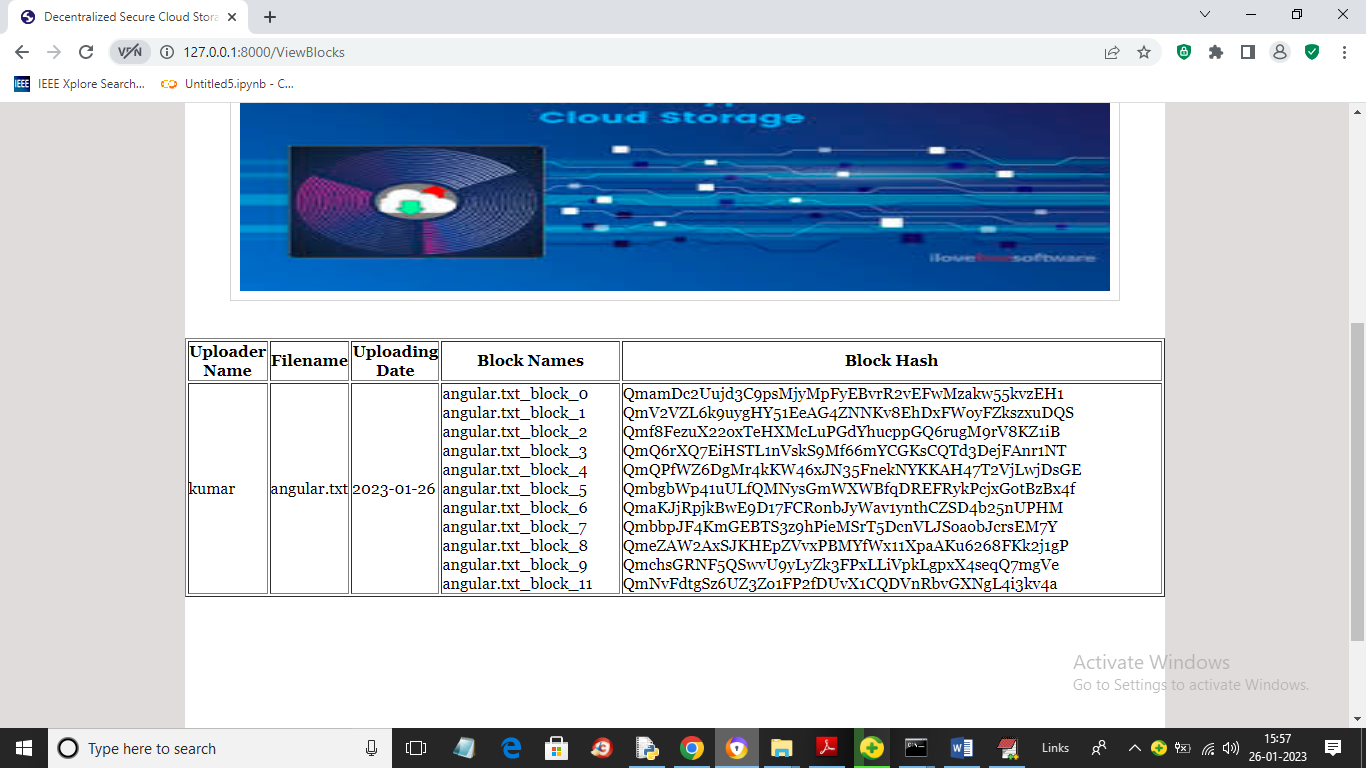
In above screen user can click on ‘Upload & Encrypted Blocks File’ link to upload file in block like below screen



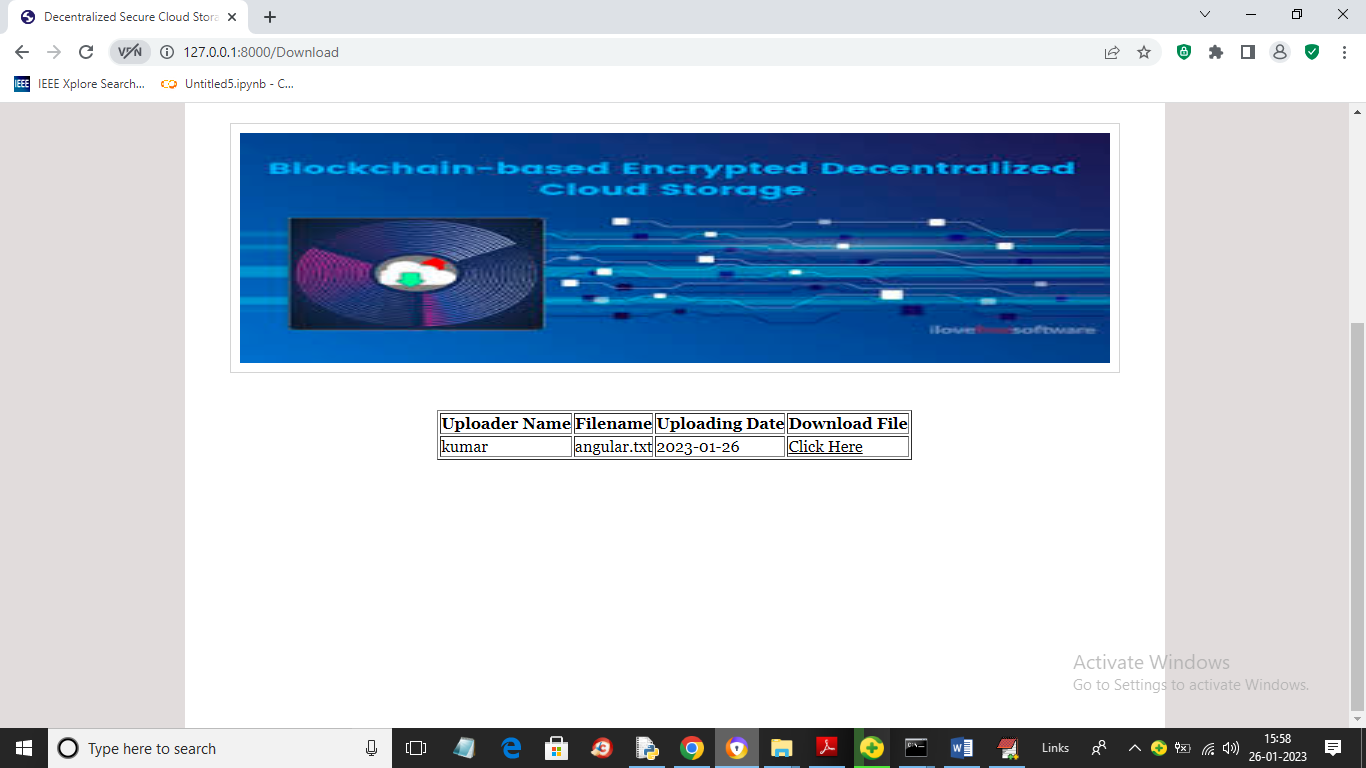
In above screen selecting and uploading file and then click on ‘Open’ and ‘Submit’ button to divide file into blocks and then encrypt each block and then save at different IPFS nodes and Blockchain and get below output



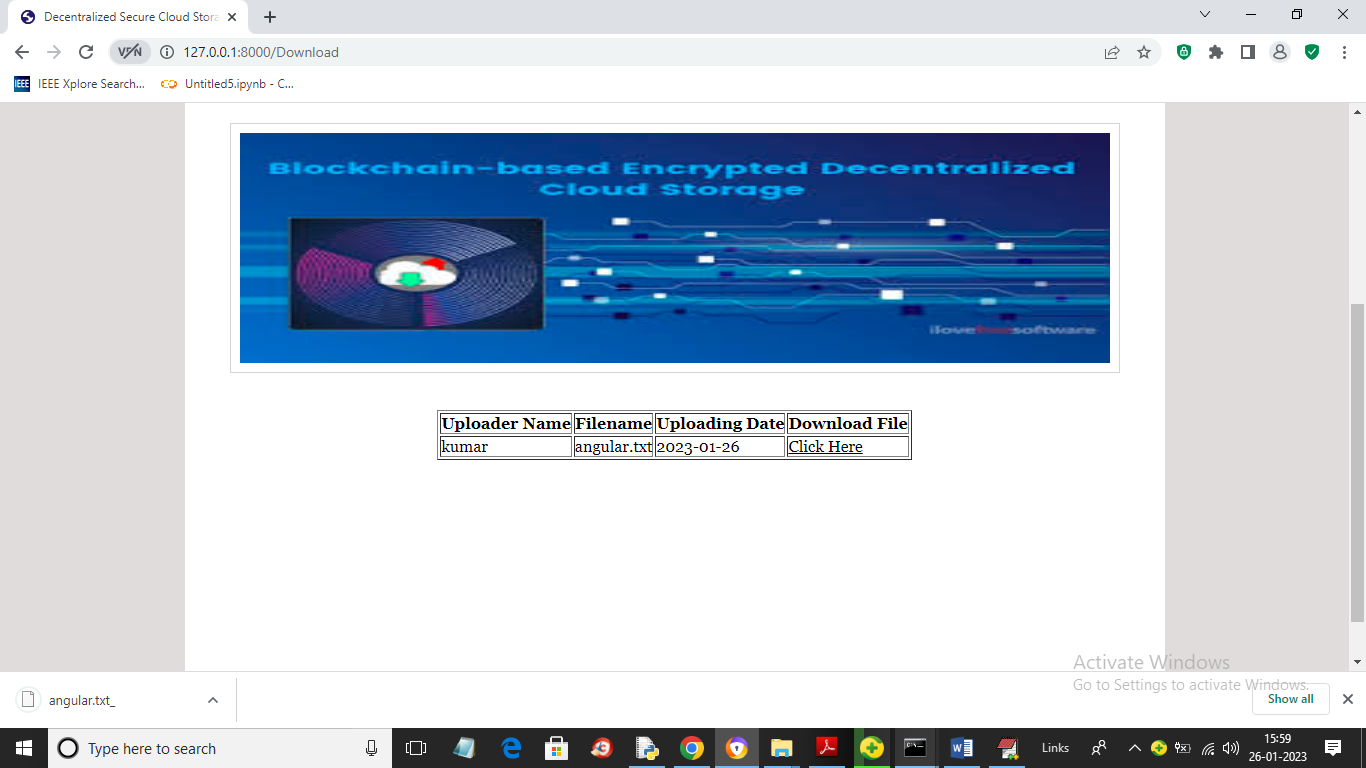
In above screen same file is divided into 11 blocks and we can see name of uploader, filename, chunk (block name), encrypted data and its hash code address of file stored in IPFS and Blockchain. Similarly you can upload other files and while uploading you need to upload small files as big files will take lots of time for encryption. Now click on ‘View Blocks’ link to get below page



In above screen for each file user can see name of file and its block name and stored address of those blocks in hash code format. Now click on ‘Download File’ link to get below screen

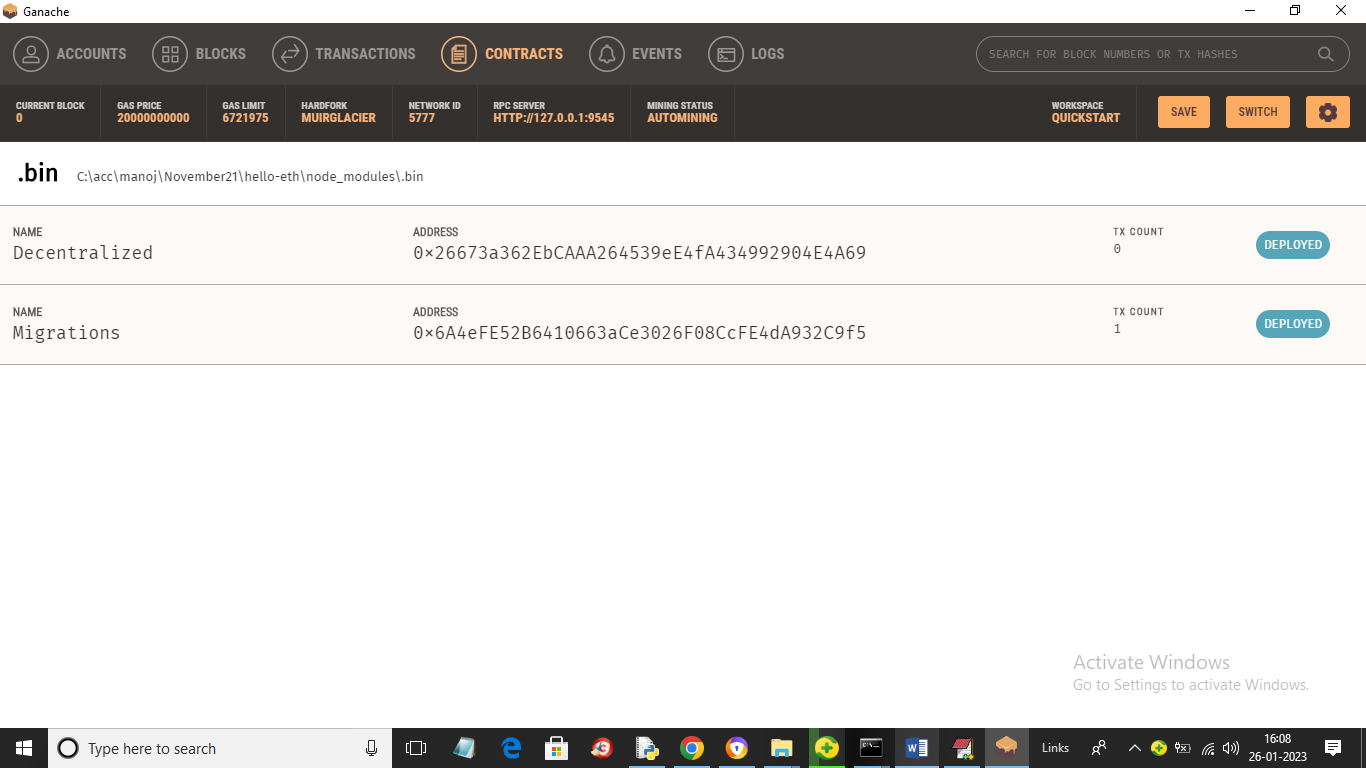


In above screen user can see list of uploaded files and then click on ‘Click Here’ link to download the file



In above screen in browser status bar we can see file is downloading and similarly you can upload and download any type of file. After downloading you can see file in encrypted format.

In Ganache also we can see Decentralized Smart Contract deployed like below screen



In above screen we can see Decentralized contract deployed