AuthPrivacyChain: A Blockchain-Based Access Control Framework With Privacy Protection in Cloud

Now-a-days all big organizations are using Cloud services to manage their application and its data storage as this cloud services provide heavy computation with cheaper cost but this advantage leads to a problem called data security as user’s data will be store at third party cloud server where malicious system admin or hacker can hack this data and misuse or tamper it. Data owner will have no control to prevent such access as his data stored on centralized cloud server which is away from his hand or control. To overcome from this problem author of this paper has introduce “AuthPrivacyChain” which uses Blockchain technology and has inbuilt data verification technique and its support immutable data which cannot be alter by anyone. Blockchain will store data in terms of transaction and each transaction is associated with verification hashcode and if any block data tamper then its hashcode get changed and verification get failed.

In propose paper author is designing access or share control by using cloud and Blockchain services. Data owner will upload file and then give access to Data user and this access details will be stored at Blockchain and encrypted file will be stored at Cloud Server. Only genuine users can request access details from Blockchain and by using those details Data user can request to cloud for file access.

In propose paper access can be provided via DIRECT and INDIRECT way. Direct means data owner authorizing some users to access his file and indirect means authorized users can give access to other users. Any time data owner can revoke or remove access from his file. After revoke no user can access his file.

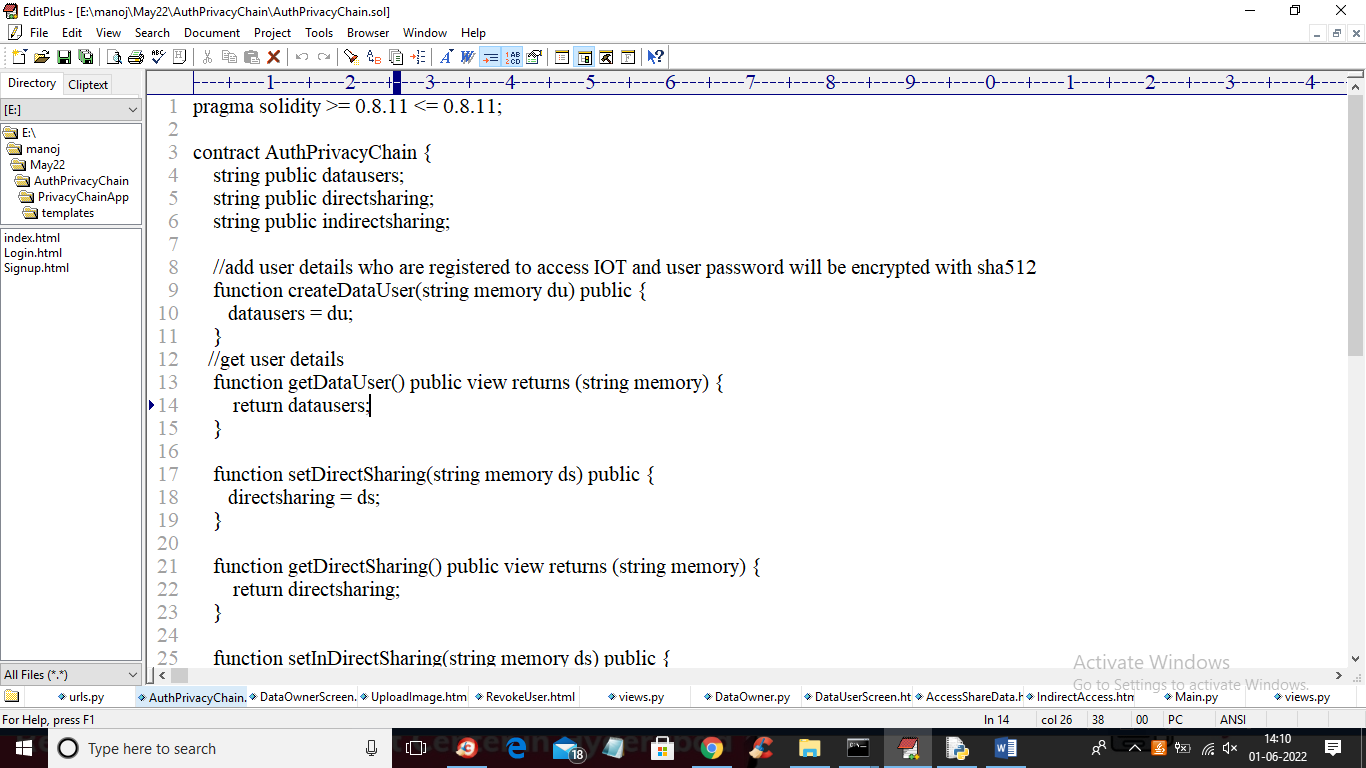
In propose paper author is designing following modules

1. Initialization: this module consist of 3 users such as data owner, data user and Cloud server
2. Registration: All users will signup with application and this details will be stored in Blockchain by using ISAVE Smart Contract function. After registration Blockchain can be used to store access permission or control. To identify each user Blockchain generate identify keys.
3. Cloud to Blockchain: cloud will issue request for registration to Blockchain
4. User to Blockchain: data owner will authorized data users to Blockchain, can upload/publish files and can perform revoke.

Here each file get encrypted using AES algorithm.

To implement this project we have shared data between two users which has role as ‘DOCTOR’ and ‘Researcher’ and data owner will give access to data user called DOCTOR or RESEARCHER. Data user can also authorized to one and other.

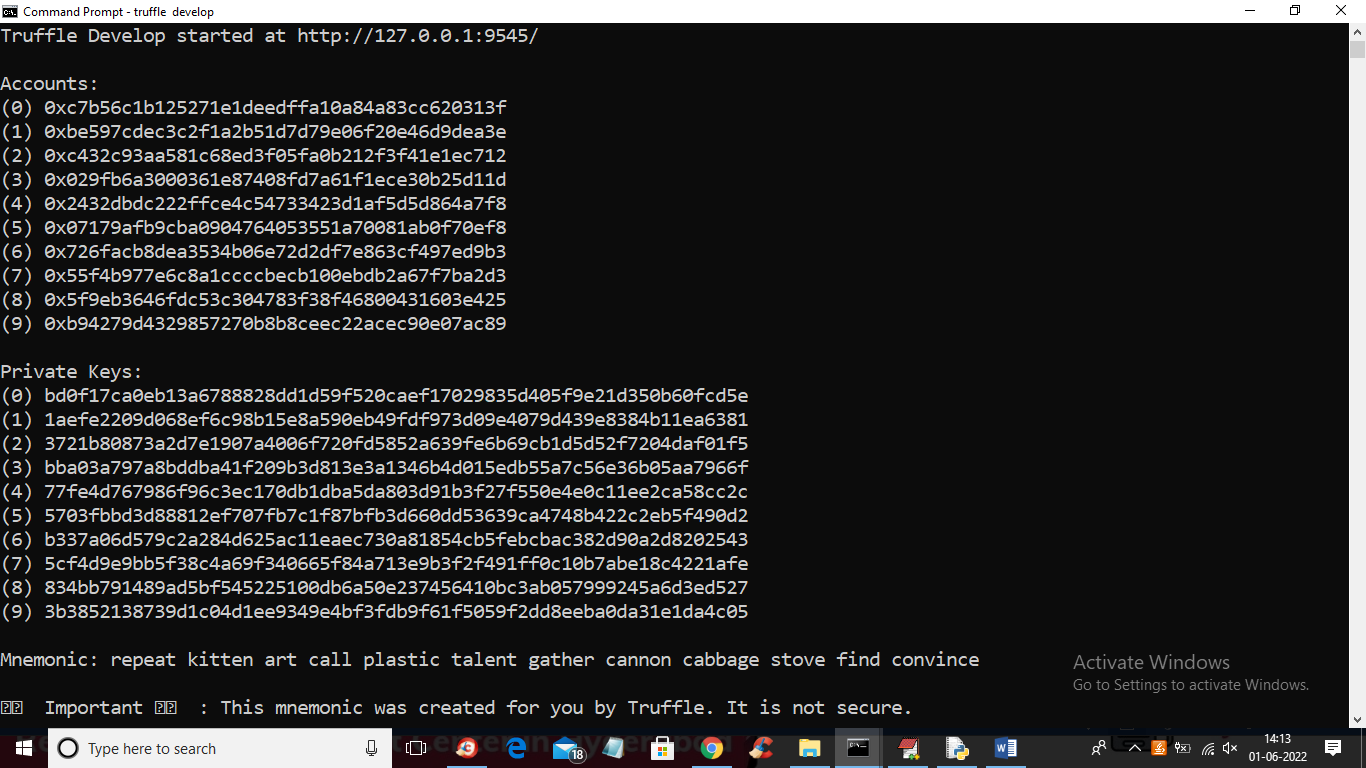
Below is the smart contract code which deployed on Ethereum tool to manage user access control and this contract is designed using Solidity code



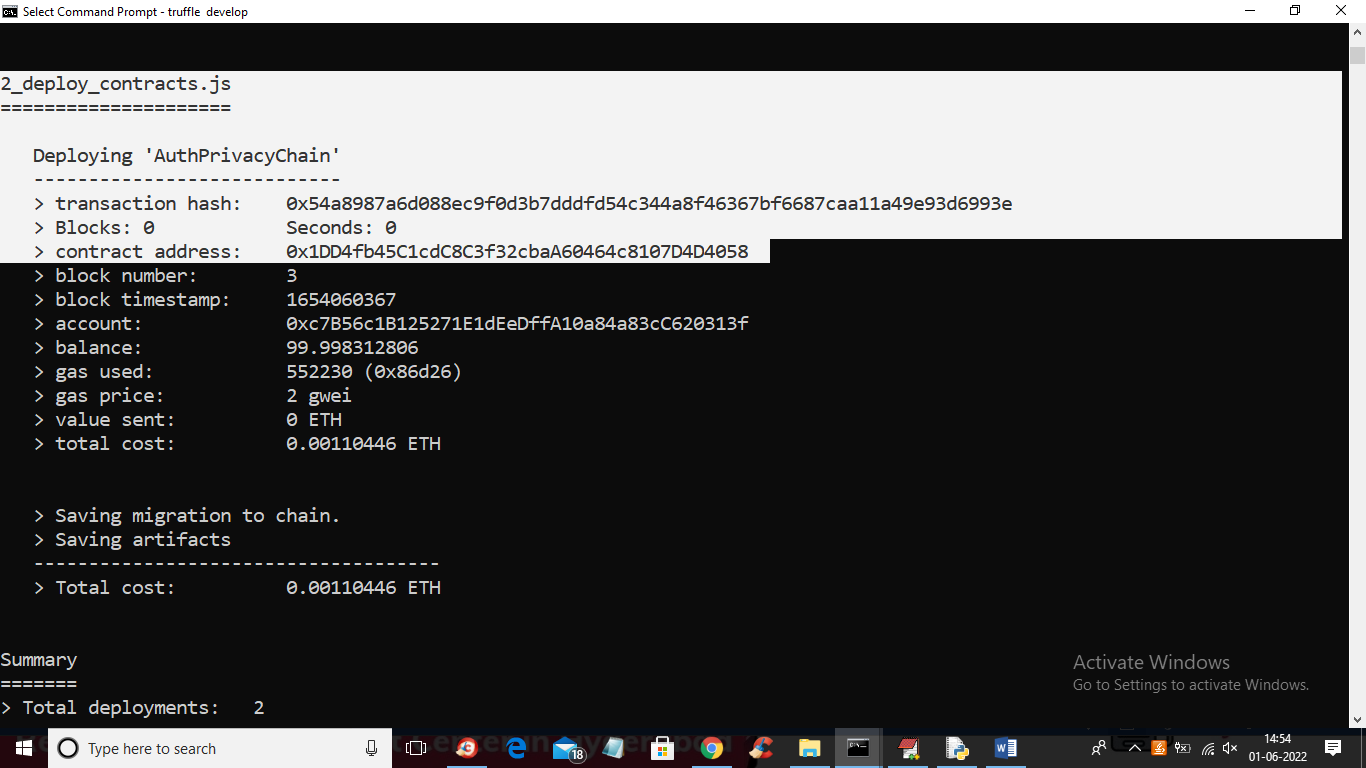
In above screen we have define function to store user details and add details for data sharing in direct and indirect way.

To deploy above contract in Blockchain Ethereum we need to follow below steps

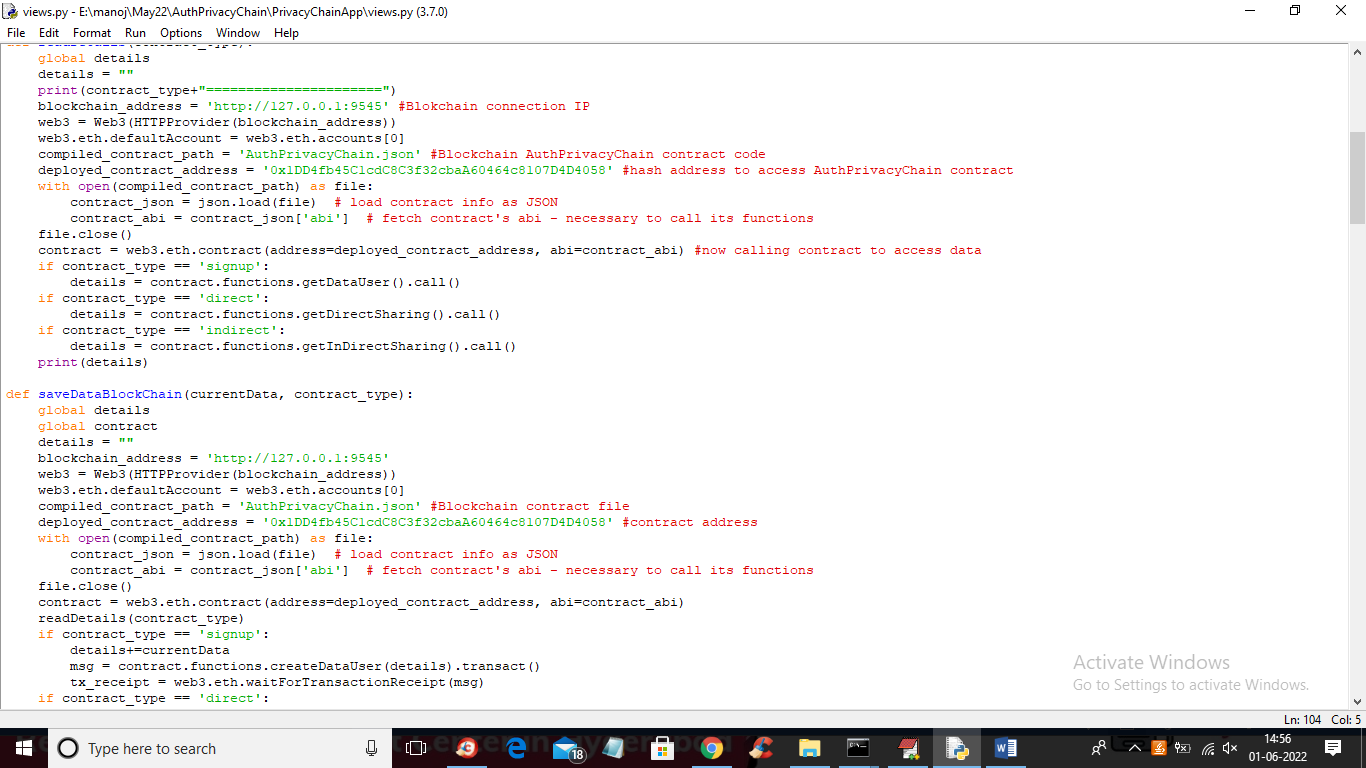
Go inside ‘hello-eth/node-modules/.bin’ folder and then double click on ‘runBlockchain.bat’ file to start Ethereum tool and get below screen



In above screen we can see Blockchain generated some default account and generated private keys and now type command as ‘truffle migrate’ and press enter key to deploy contract on Blockchain Ethereum and will get below output



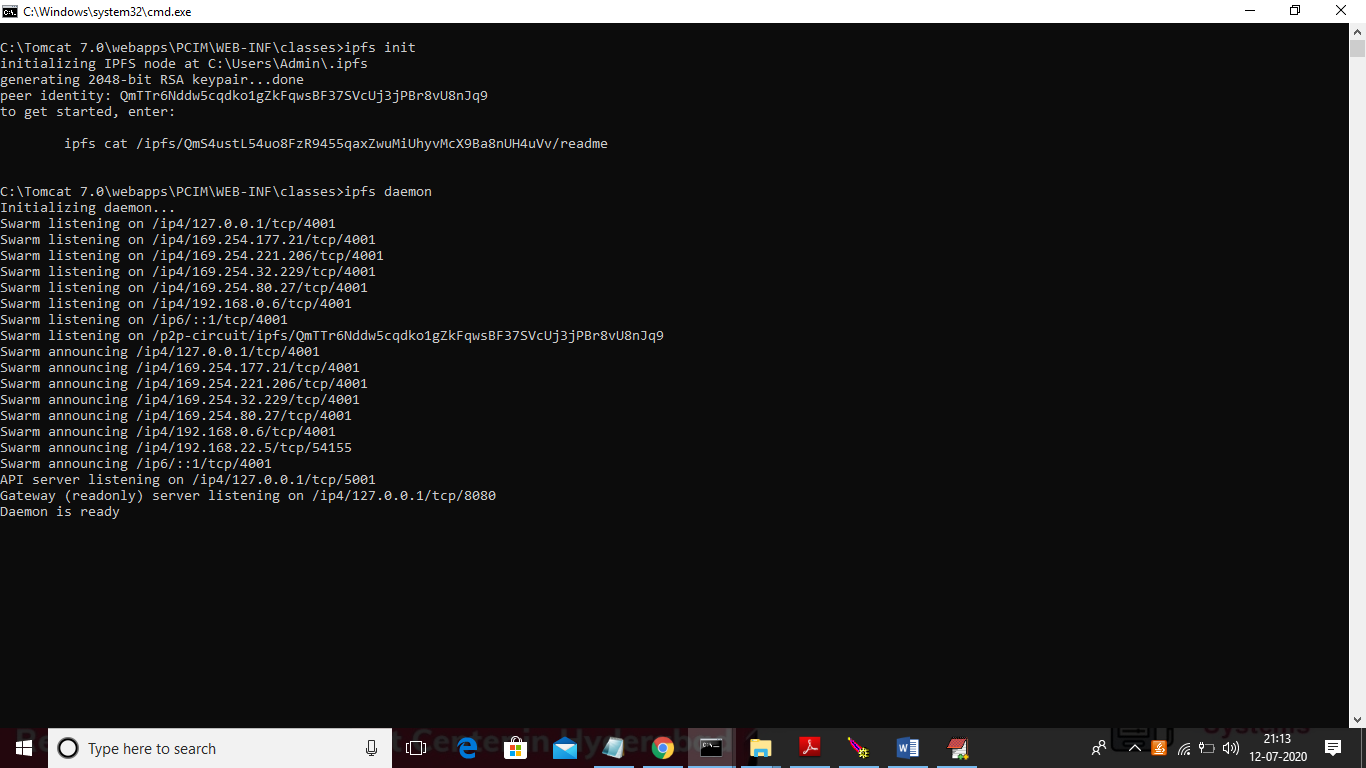
In above screen in white colour text we can see ‘AuthPrivayChain’ contract deployed and we got contract address also and this address we will specify in Python program to store and retrieve data. In below python code you can see Ethereum smart contract calling code



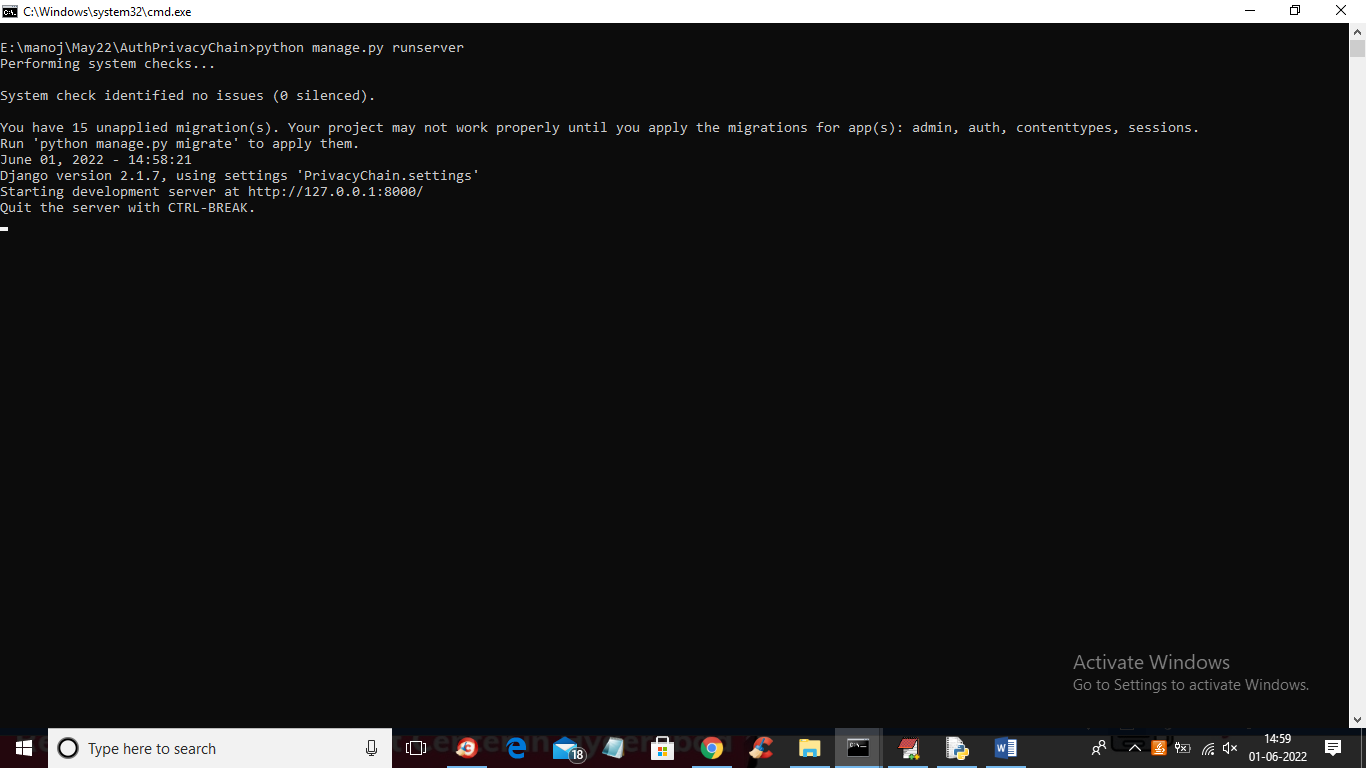
In above screen read red colour comments to know how to call contract.

SCRCEEN SHOTS

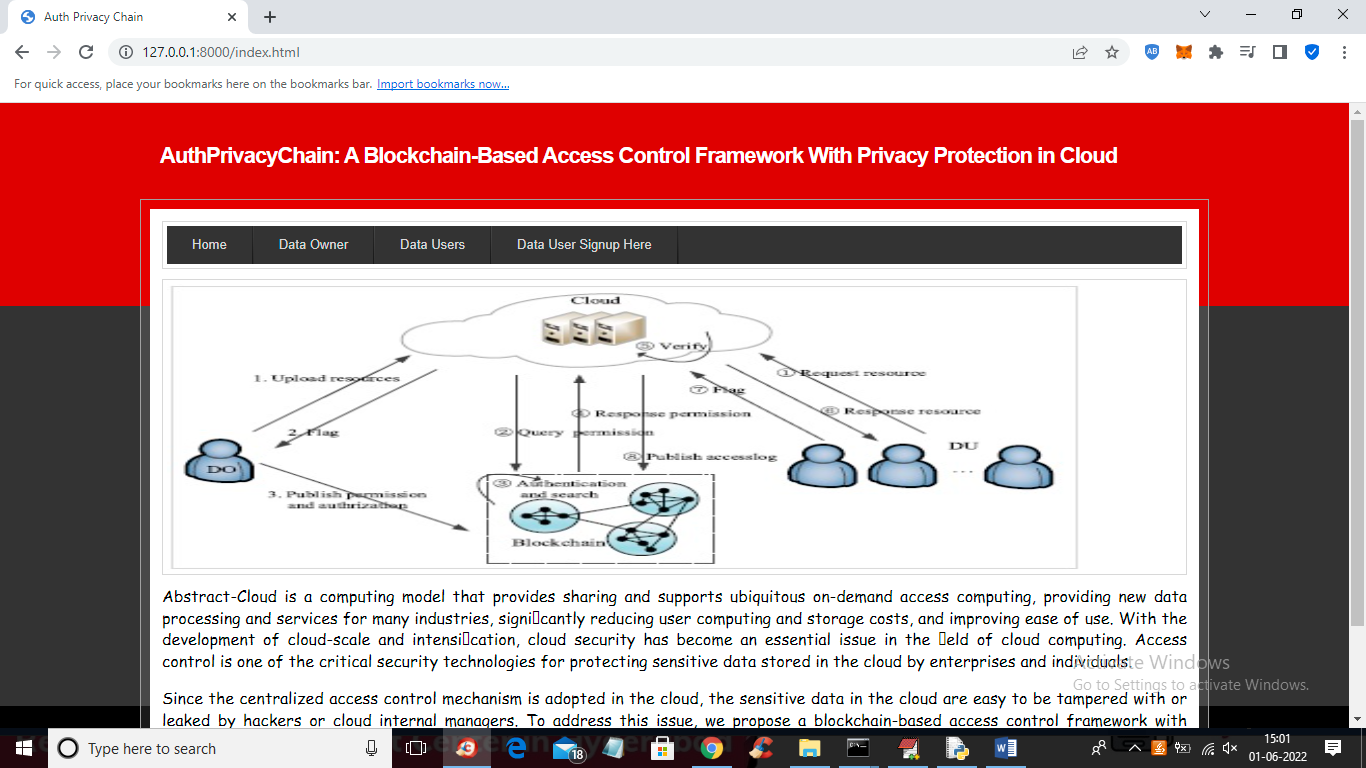
First double click on ‘Start\_IPFS.bat’ file to start cloud server and get below screen



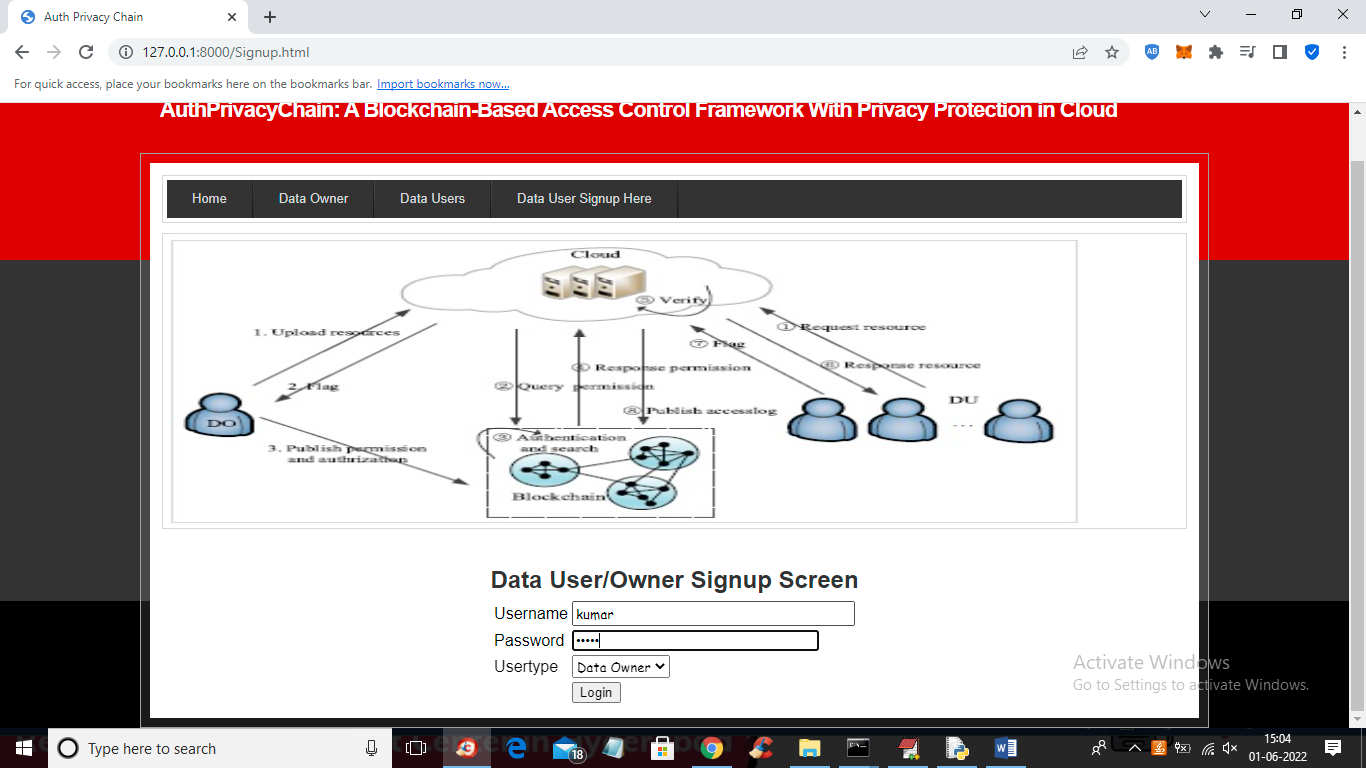
In above screen cloud server started and now double click on ‘runServer.bat’ file to start python server and will get below screen



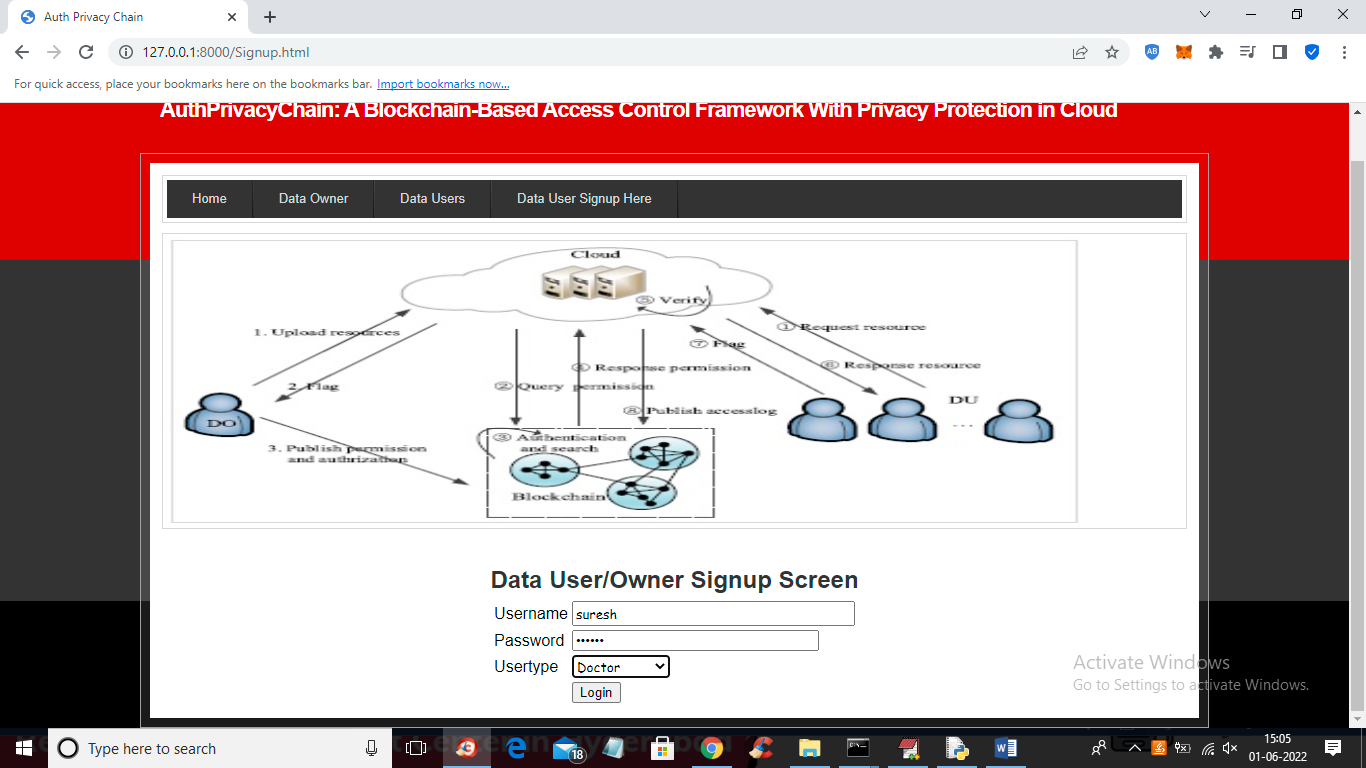
In above screen python Web Server started and now open browser and enter URL as ‘http://127.0.0.1:8000/index.html’ and press enter key to get below screen



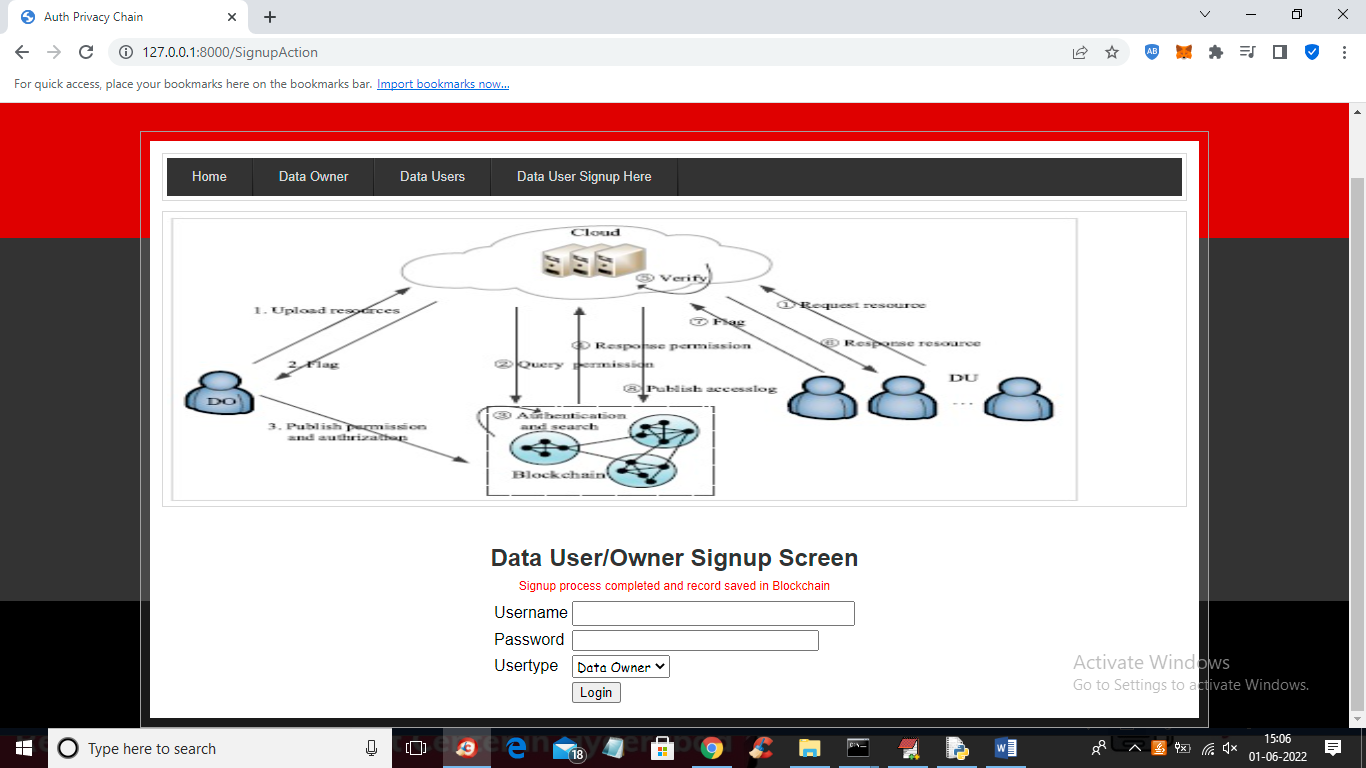
In above screen click on ‘Data User Signup Here’ link to add different users such as Data Owner, Doctor and Researchers



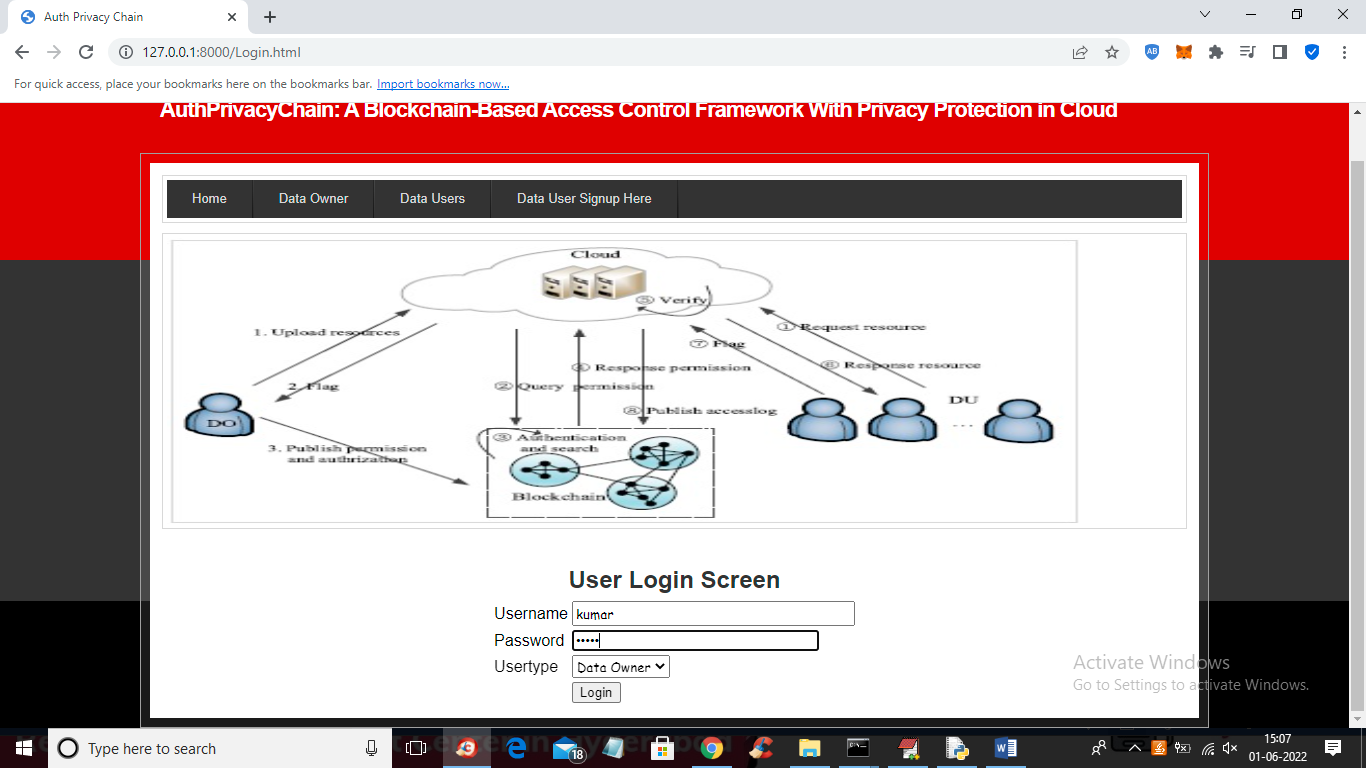
In above screen data owner is signing up and press button to complete signup process and similarly add doctors and researchers



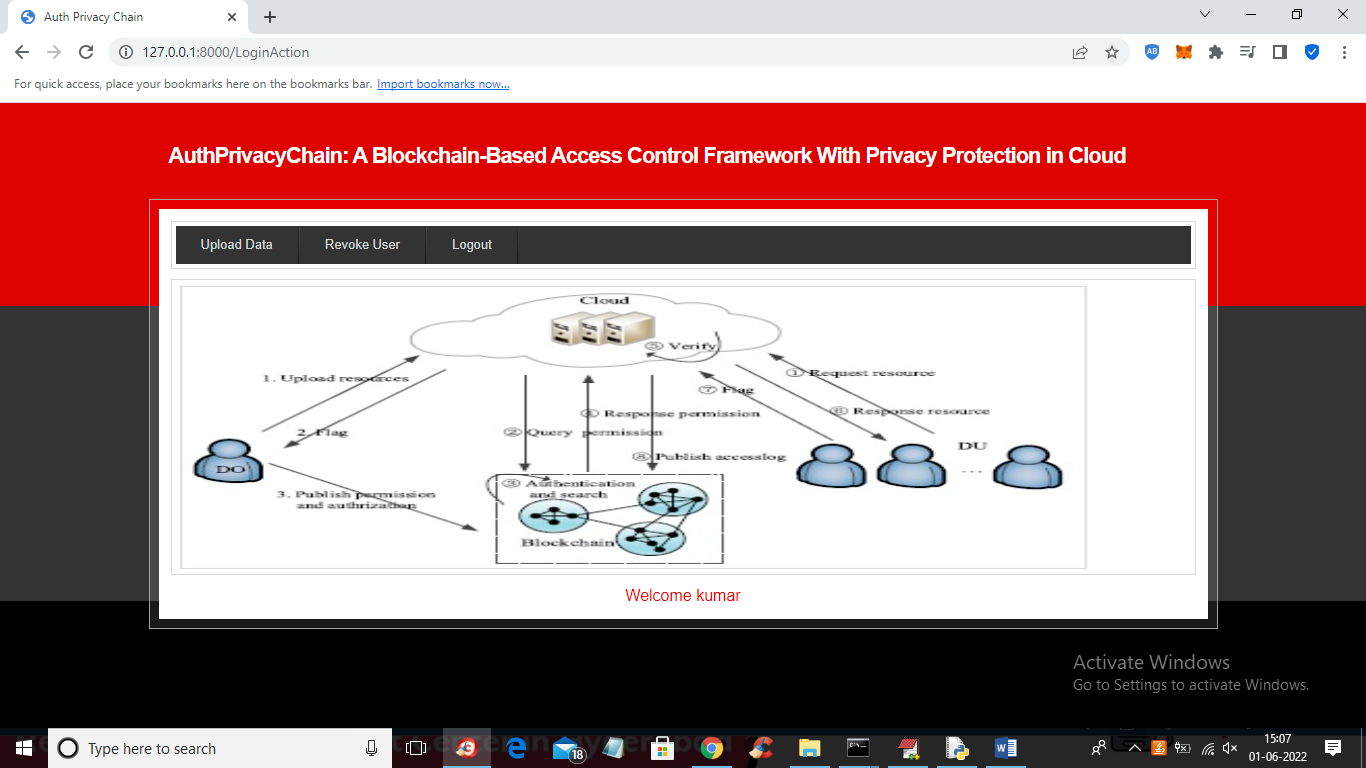
In above screen doctor is signing up



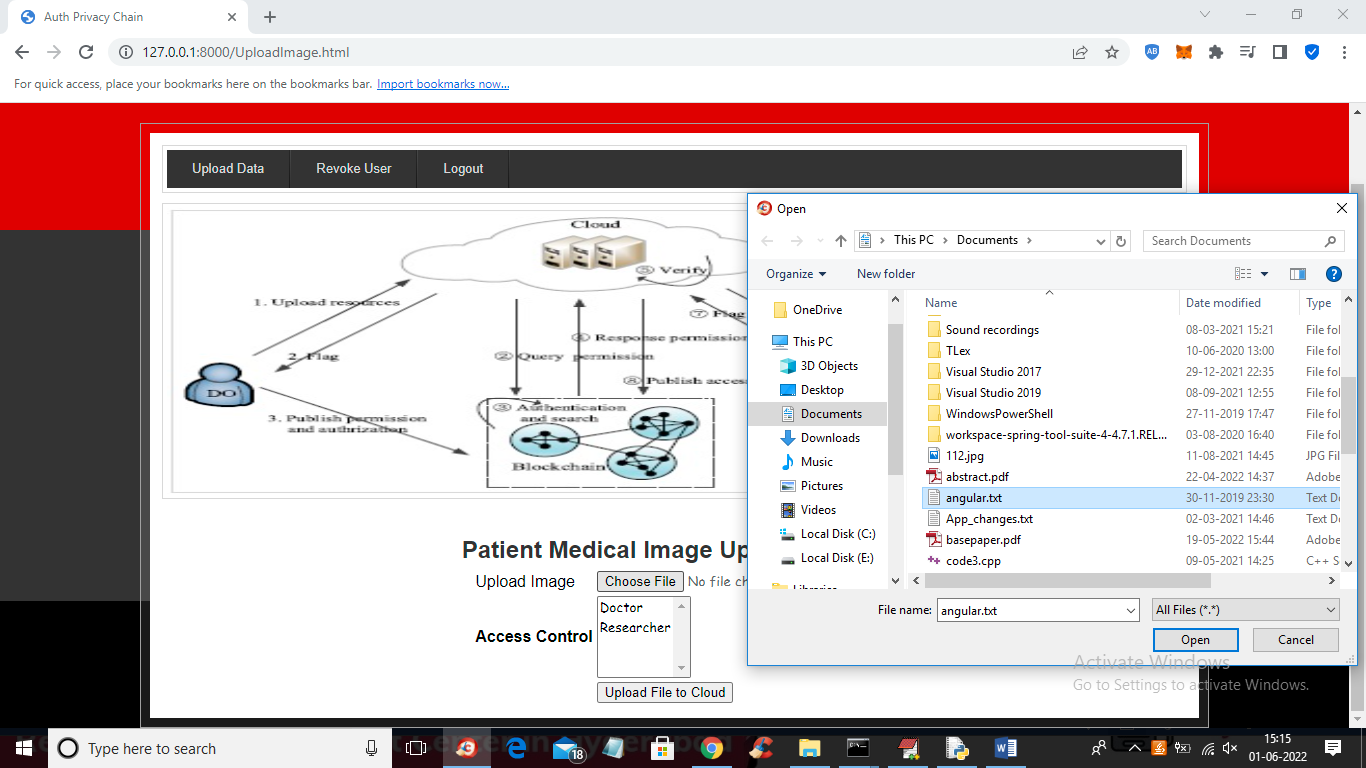
In above screen signup process completed and now click on ‘Data Owner’ link to login as data owner

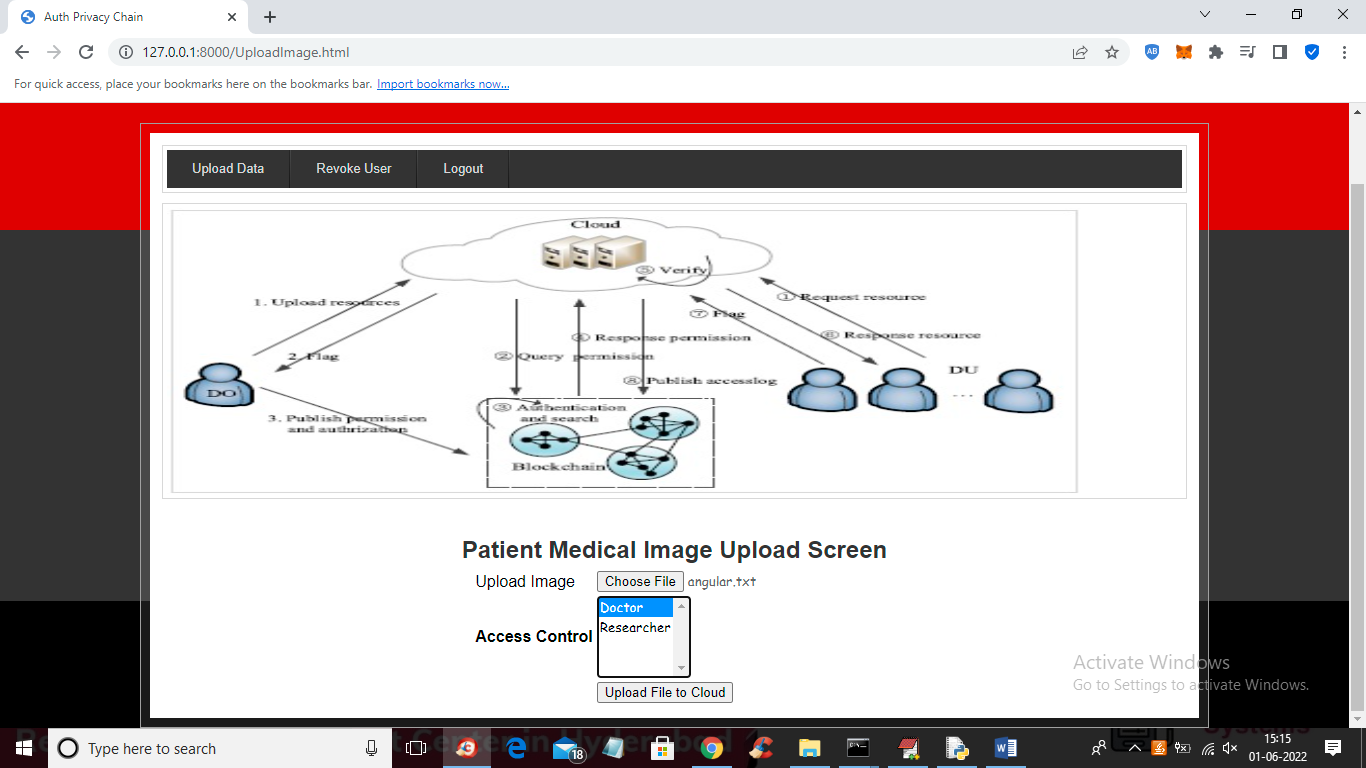


In above screen data owner is login and after login will get below screen

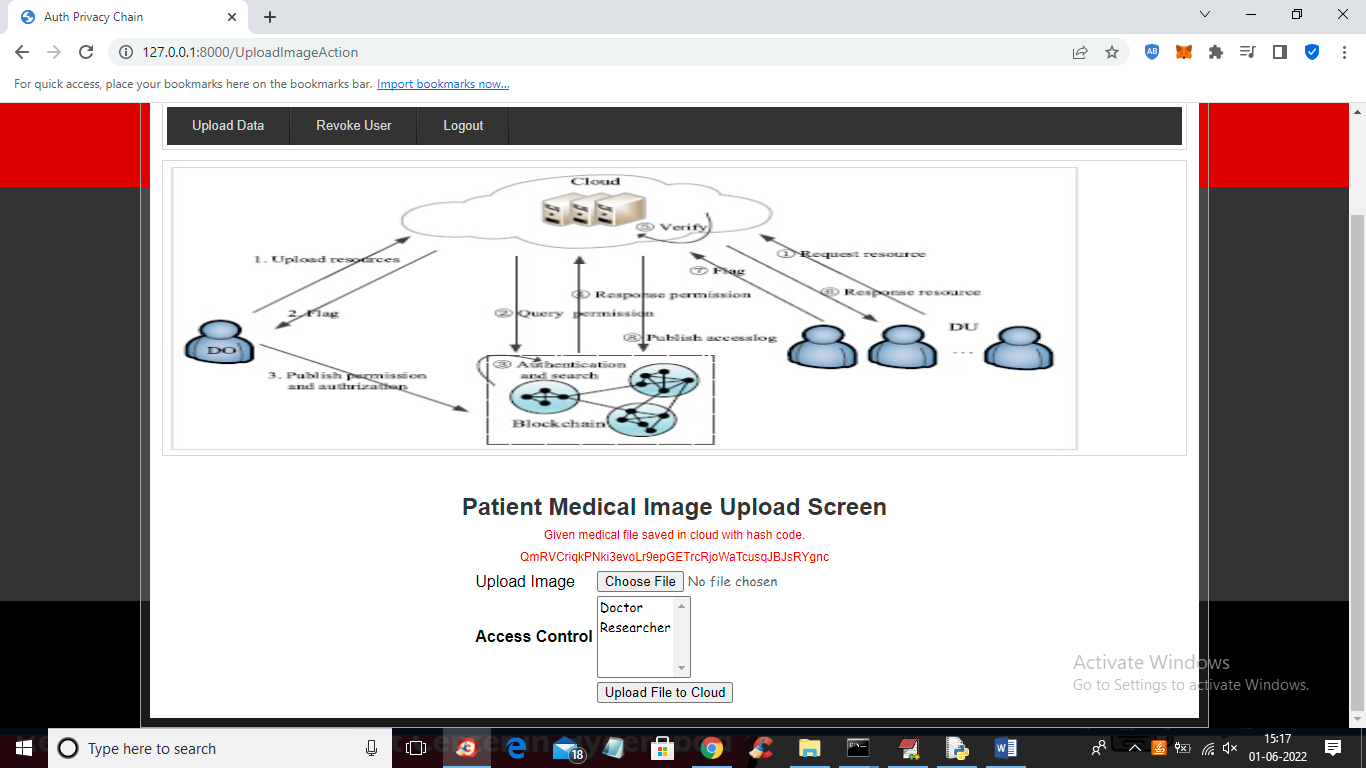


In above screen data owner can click on ‘Upload Data’ link to upload file to cloud in encrypted format

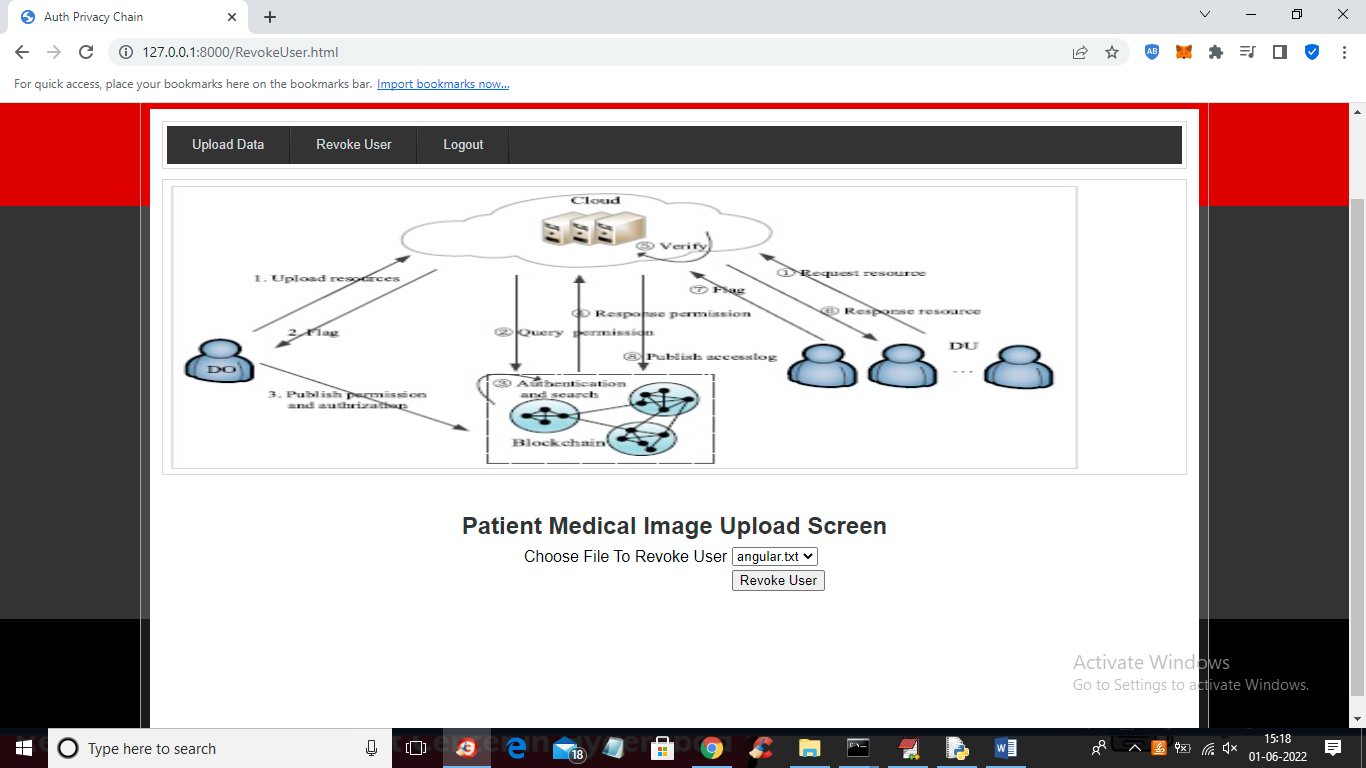




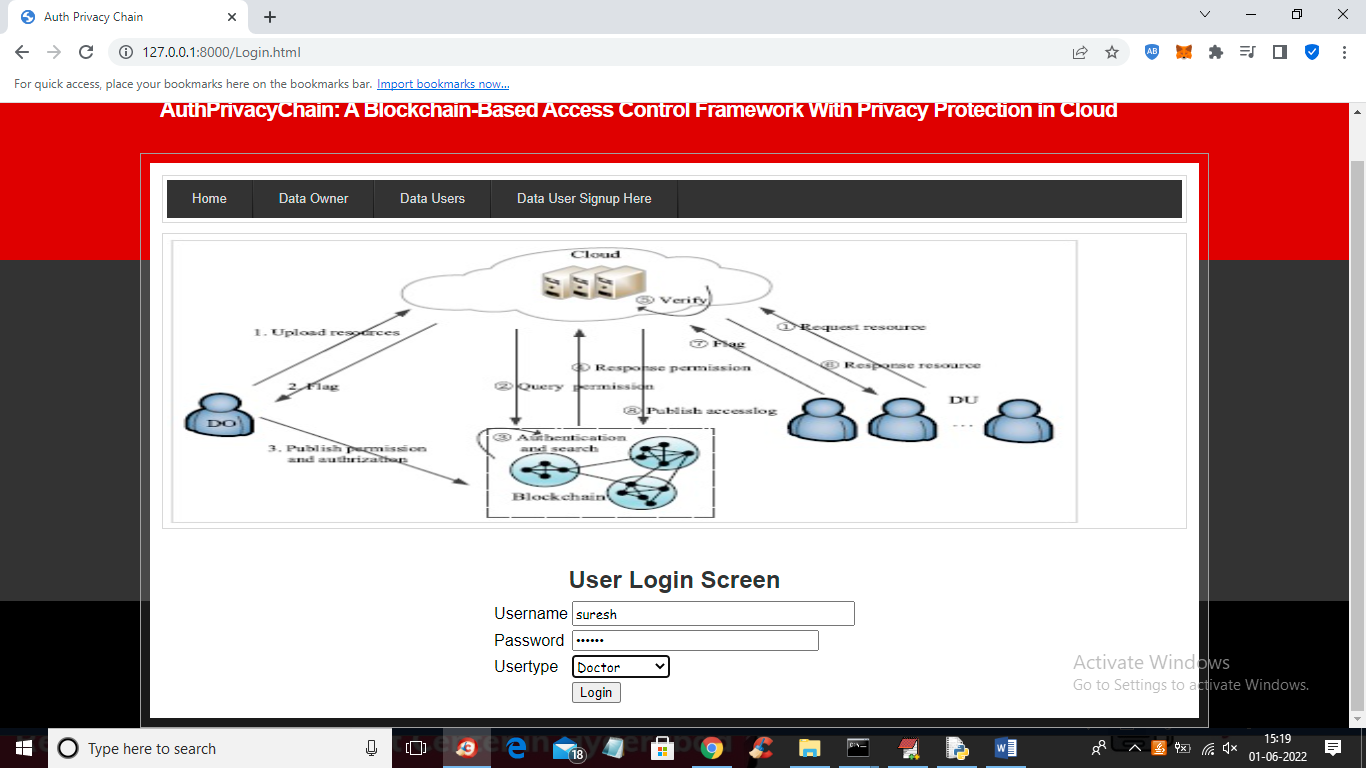
In above screen data owner uploading file and then select the access user as doctor or researcher or both and press button to upload file and in above screen I am giving permission to user ‘doctor’ and press button to get below output



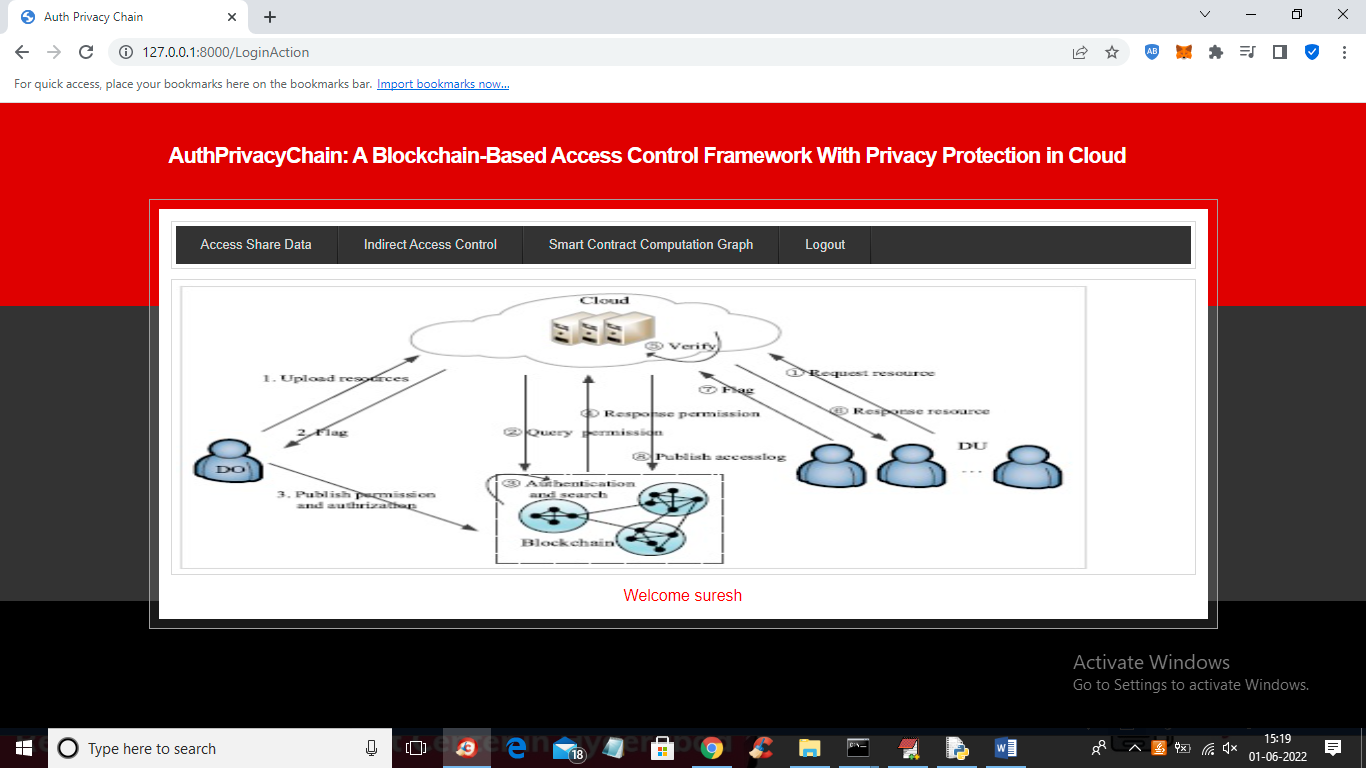
In above screen file is uploaded and we can see the storage hash code and similarly data owner will click on ‘Revoke user’ link to remove access from file by selecting it



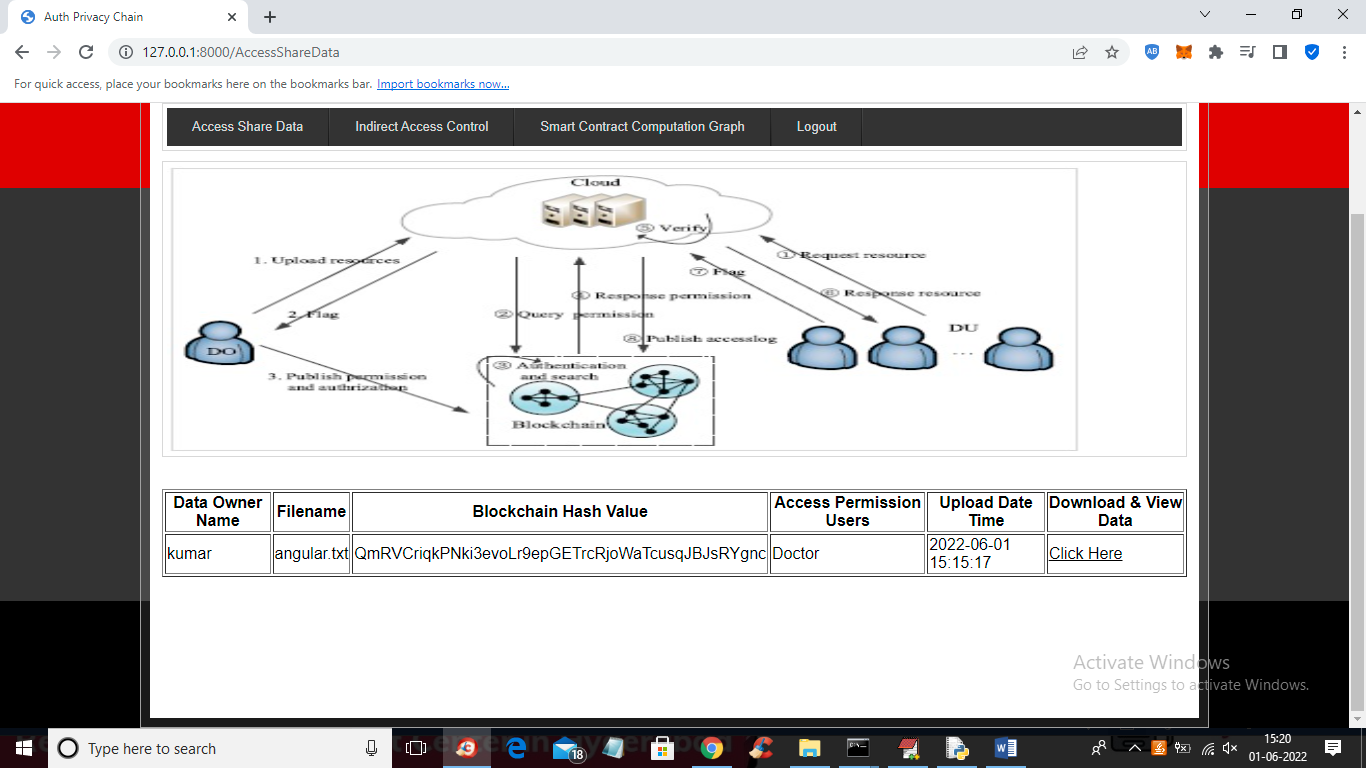
In above screen select any file and press button to remove access from it and now logout and login as ‘doctor’ to see access is available or not



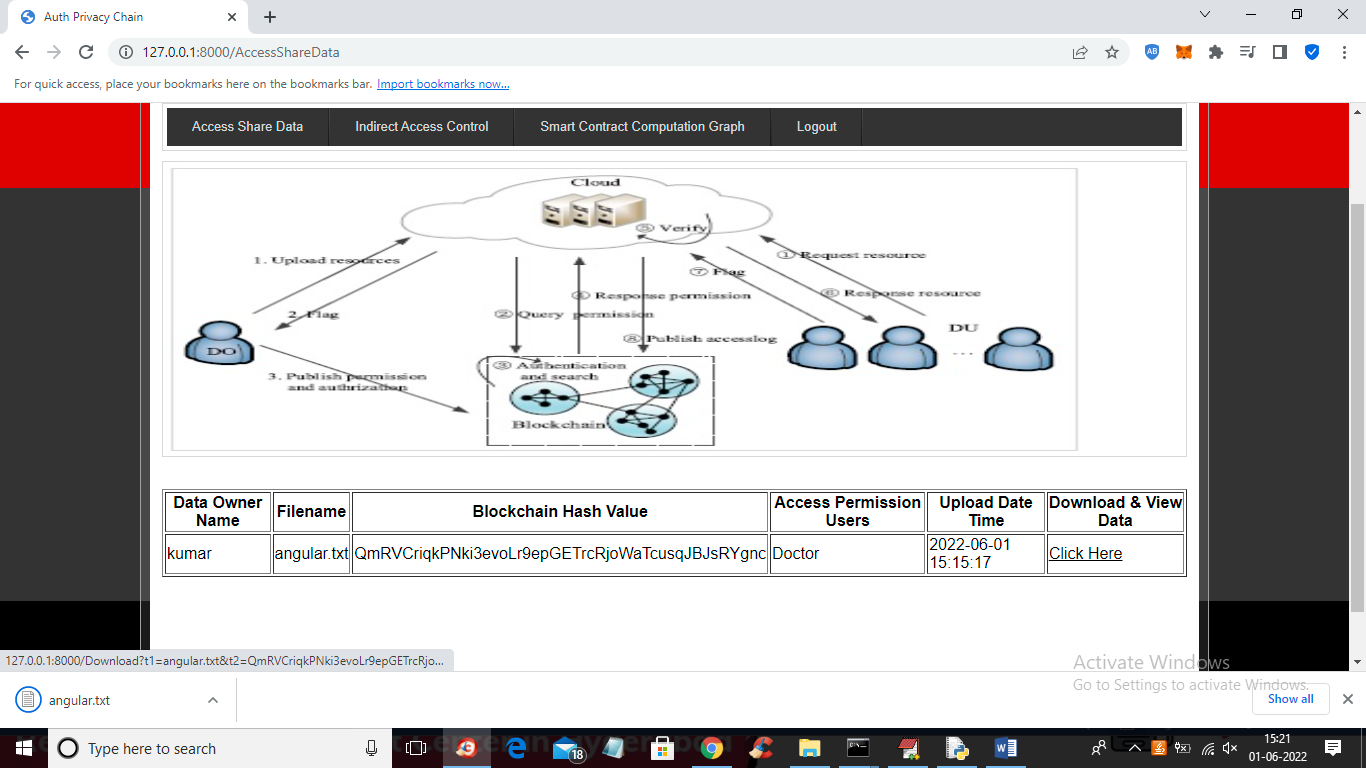
In above screen doctor is login and after login will get below screen



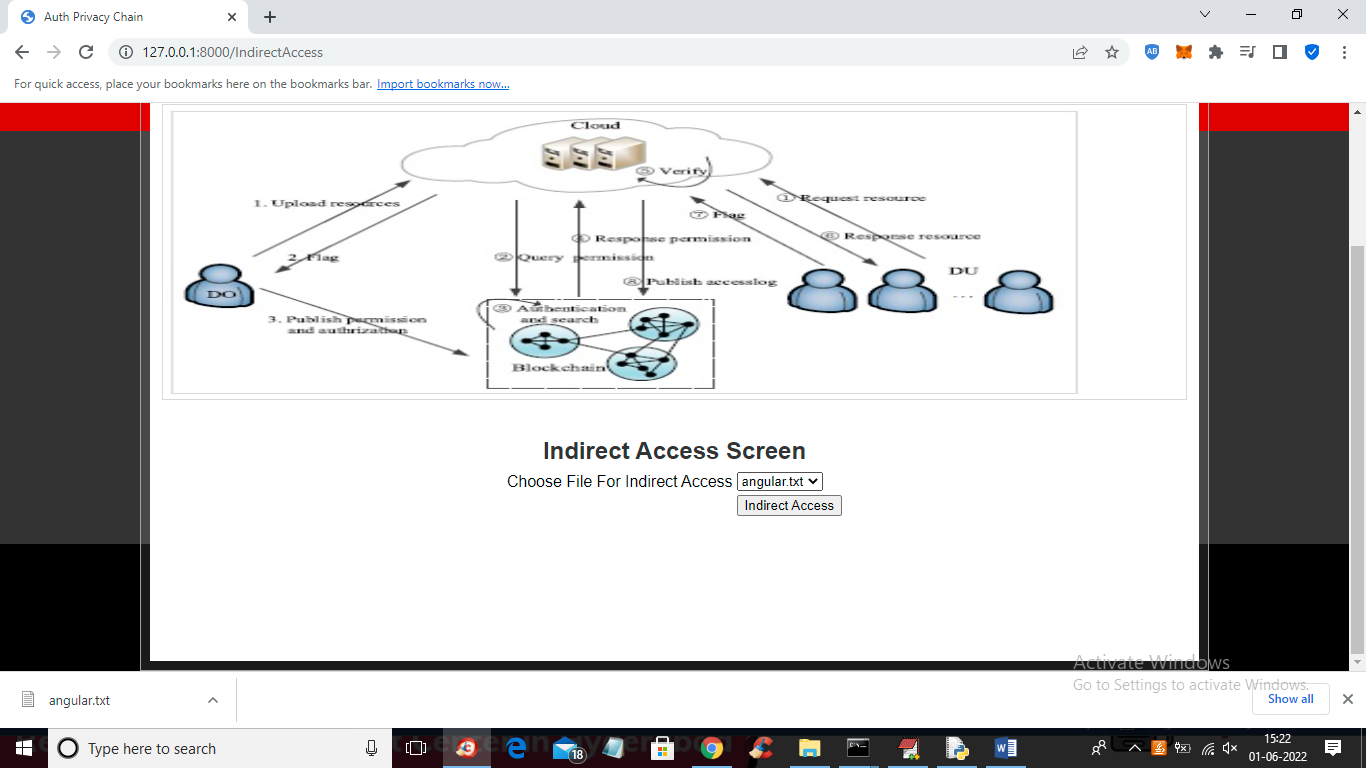
In above screen doctor can click on ‘Access Share Data’ link to view all files shared by data owner



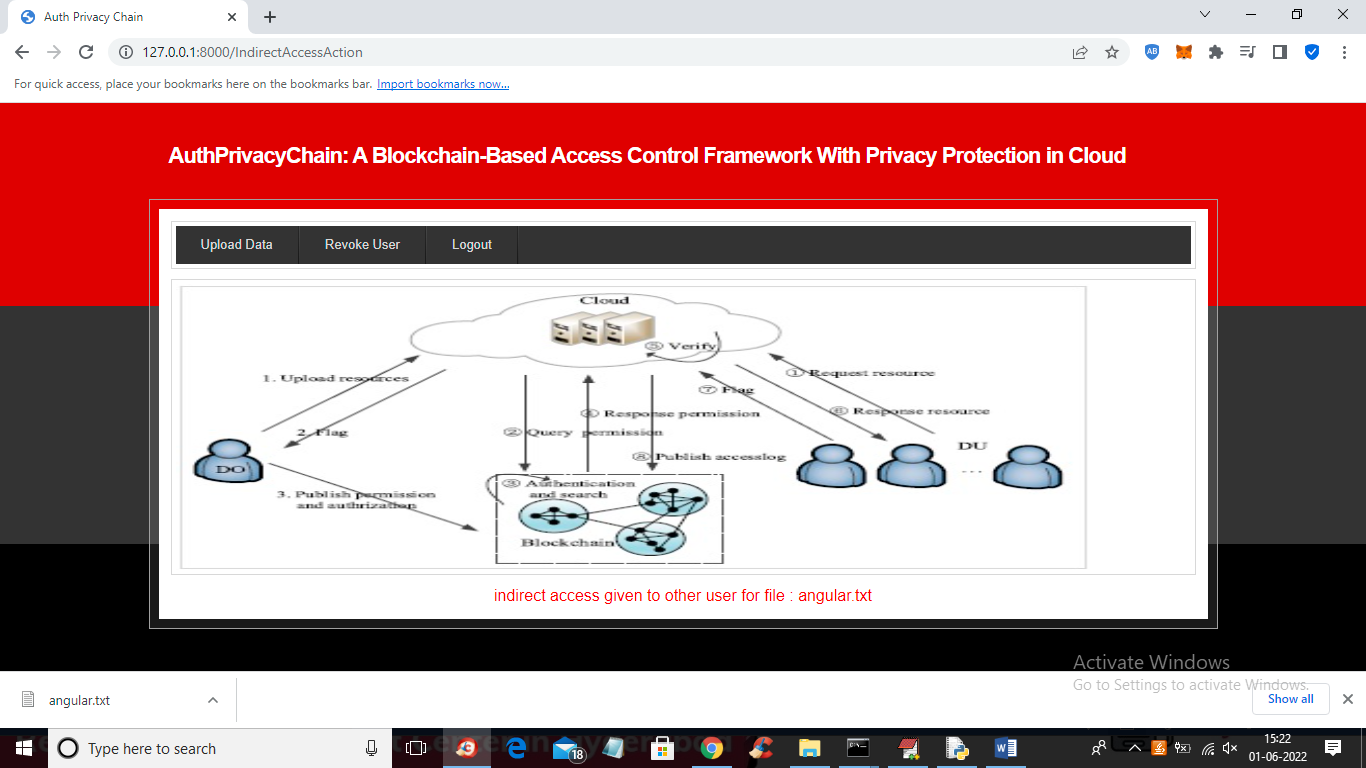
In above screen doctor can view all files shared by data owner and can click on ‘Click Here’ link to download that file



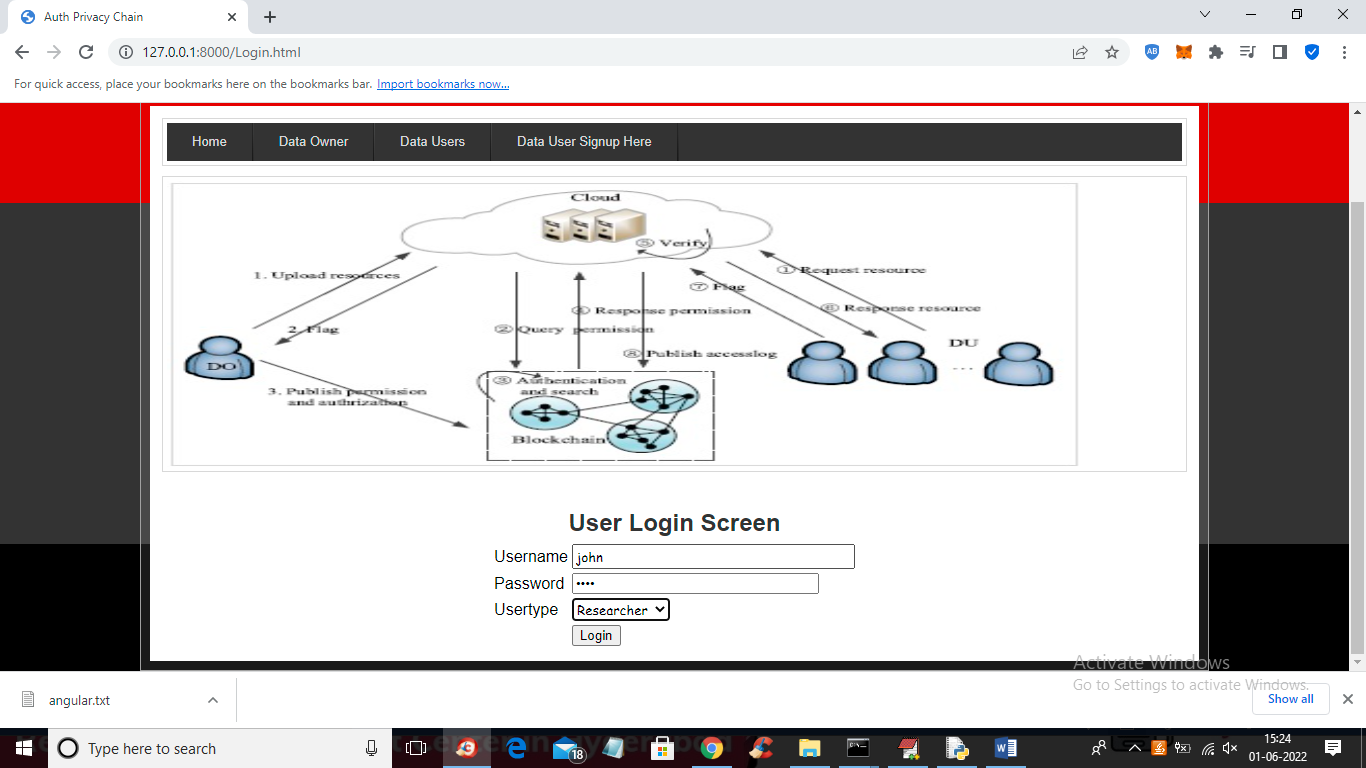
In above screen in browser status bar we can see file is downloading and now doctor can give access to this file to researcher by clicking on ‘Indirect Access Control’ link



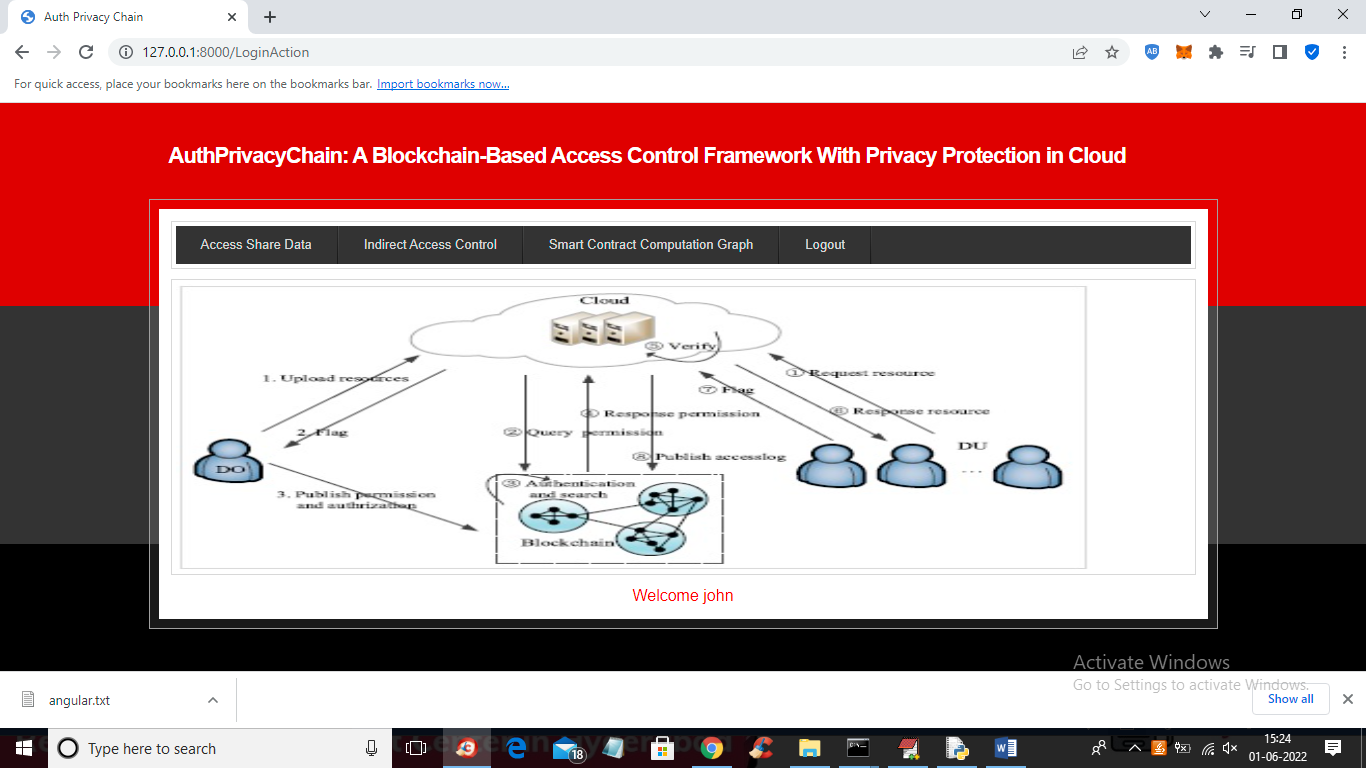
In above screen doctor can select file and click button to give access to researcher and get below output



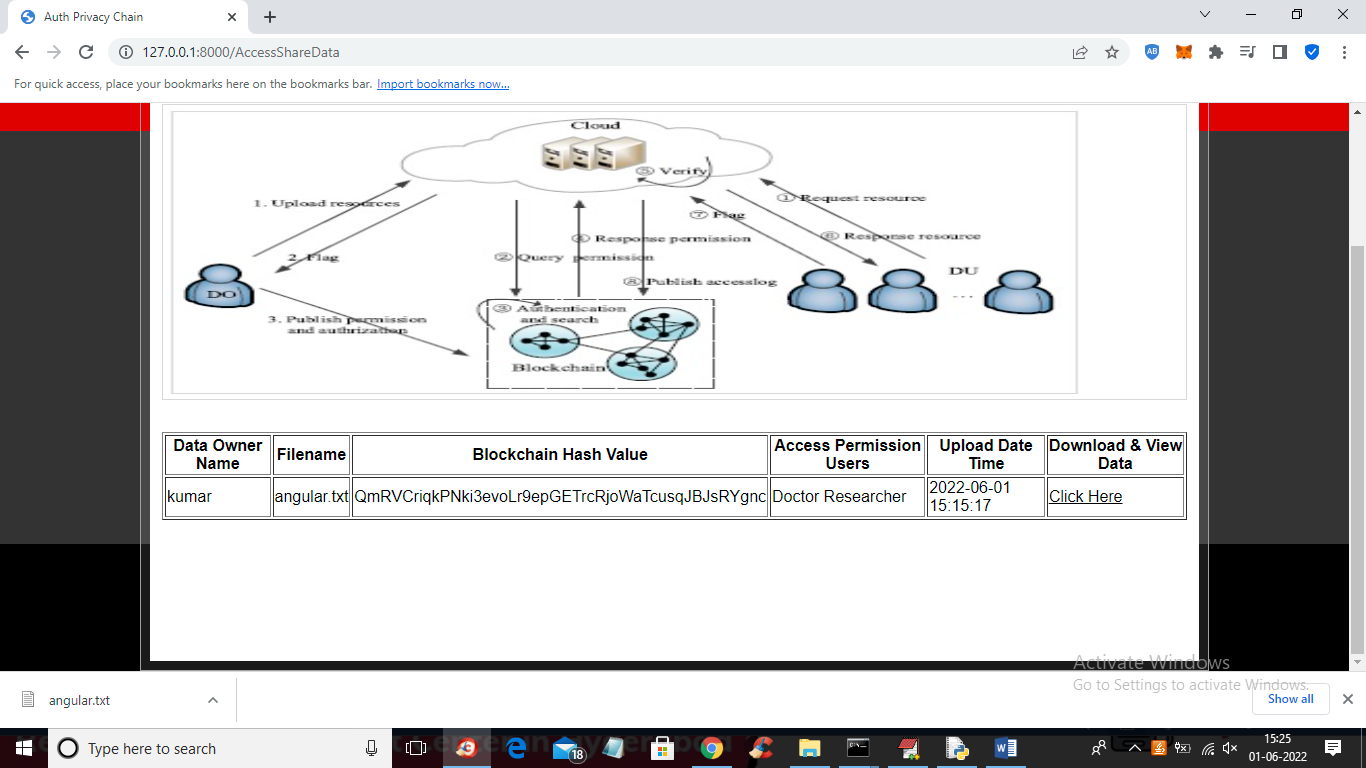
In above screen indirect access given for file ‘angular.txt’ and similarly logout and login as researcher to access that shared file



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



In above screen researcher can access file by clicking on ‘Access Share File’ link and get below output



In above screen we can see researcher also got access for file and now click on ‘Smart Contract Computation Graph’ link to get below graph



In above graph x-axis represents filename and y-axis represents computation time to encrypt and store file in Blockchain. Similarly you can add nay user and can share and revoke data between them