A Secure and Cost-Efficient Blockchain Facilitated IOT Software Update Framework

IOT’S sensors are using everywhere such as Patient Body temperature monitoring, road side traffic monitoring, agriculture field temperature monitoring and many more. This IOT will sense its deployed environment data and then report to centralized server for further processing. This IOT need to update its software’s to fight against attack as malicious attackers may hack this IOT to report wrong information to centralized server.

In all existing techniques IOT were receiving updates from single centralized server which can get failed due to any reason and in such situations updates will not be available.

Existing Techniques contains many other attacks listed below

1. IOT manufacturer: malicious IOT manufacturer can send wrong or old updates to get money from IOT owner
2. IOT owner: this are the IOT owners who purchase IOT from manufacturer and then can generate fake payment to receive updates from the manufacturer as all existing payment are based on traditional HTTP REST API based processing which can be easily hack
3. Malicious Attacker: In this attack, a malicious attacker aims to interrupt the software notification sent by the manufacturer to IOT owners, causing delays/failures of software updates.
4. Confidentiality attack: In this attack, a malicious attacker attempts to retrieve the contents of a software update during its delivery process with the aim of receiving a software update it should not have while also avoiding payment.
5. Invalid update attack: In this attack, a malicious attacker attempts to send an invalid/damaged software update to IOT devices aiming to damage the functionality of the IOT device.
6. Roll-back attack: In this attack, a malicious attacker aims to send a valid old software with a known pre-existing vulnerability that the attacker can exploit to damage the functionality of the IOT device

To overcome from above attacks author employing Blockchain based IOT software’s updates which consists of following technologies

1. Blockchain: Blockchain has inbuilt support for data security and verification and store all data as Block or transaction and associate each block with unique hash code. Blockchain verify hash code of each block storage and if hash code does not match then it will detect as data alteration and this verification make Blockchain as tamper proof and impossible for data change. Blockchain can be access by using smart contract which contains function to store or retrieve software updates
2. CPABE: cipher policy attribute based encryption is applied to encrypt software updates. Each encryption technology is based on key generation for encryption and decryption. This key generation required heavy computation so generating keys for each IOT require more space and computation and this computation can be reduced by applying CPABE algorithm. CPABE uses list of allowed users or IOT as input and for all users generate single keys and the users or IOT who are in list are allowed for data decryption. So manufacturer add list of paid IOT and then generate keys and only those IOT can receive and decrypt software updates.
3. Elliptic Curve Digital Signature Algorithm (ECDSA): software updates sent by manufacturer must be check for attacks so ECDSA will generate hash code on software update and sent to IOT and then IOT will also generate hash code on received updates and if generated and received updates match then IOT consider received software is attack free. If received data alter by attacker then different hash code will be generated and attack will be detected.
4. IPFS storage: Interplanetary File System is used to store all encrypted software updates sent by Manufacturer. As Blockchain contains small amount of memory and its storage is also very costly so author using IPFS for software update storage. Blockchain will encrypt software update and then send to IPFS for storage and this IPFS will send back address of store data which will be used by Blockchain to locate that store data in future.

So by using above technologies we can avoid all attacks in the IOT software updates network.

Extension Concept

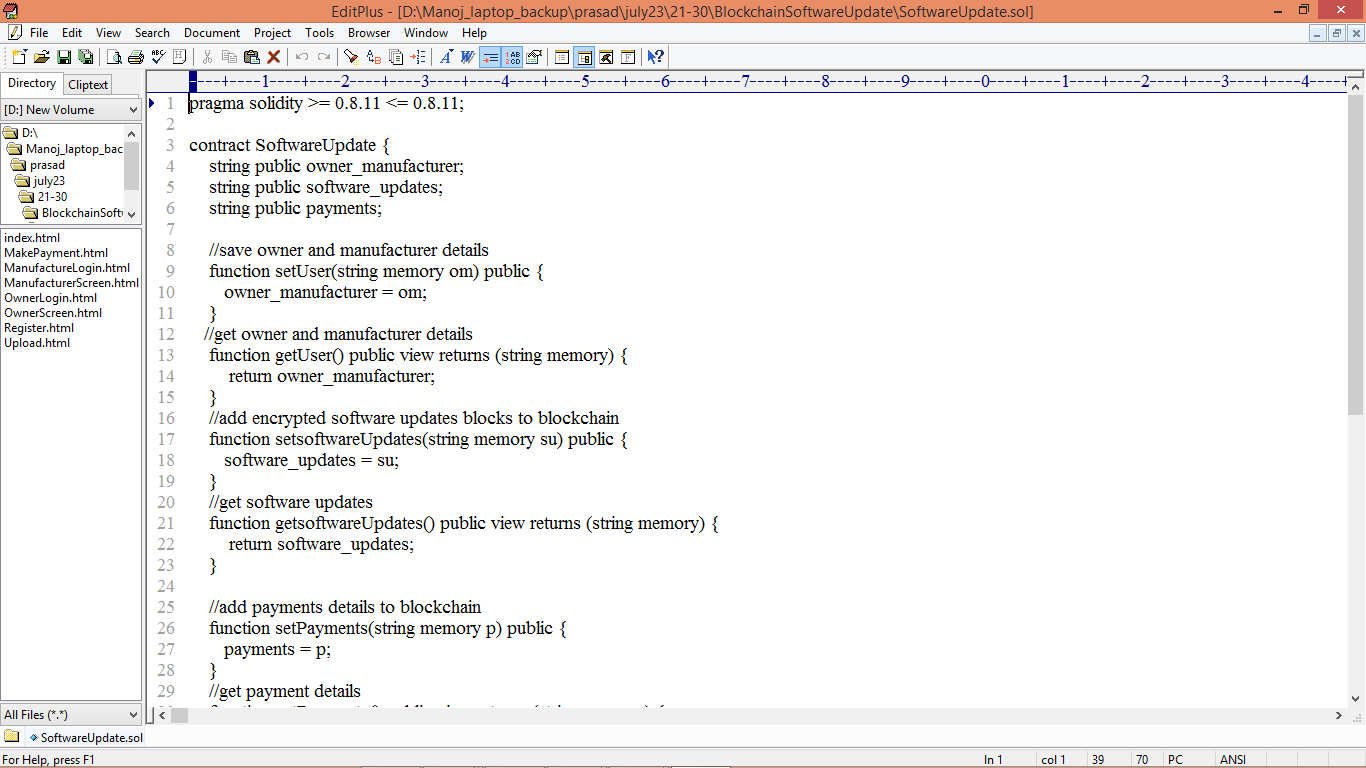
In propose work author storing all software updates in single IPFS location and if this IPFS hack then all updates will get leak or corrupted. In Extension work will divide software updates file into Blocks and then encrypt each block using CPABE algorithm and then store each block in IPFS server at different nodes in different address. IPFS will return memory address of stored block and this address will be saved in Blockchain. While downloading software updates 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 used by the IOT devices.

By storing data in different IPFS location we cannot allow attackers to know the address of all file blocks and he cannot download or alter software updates file.

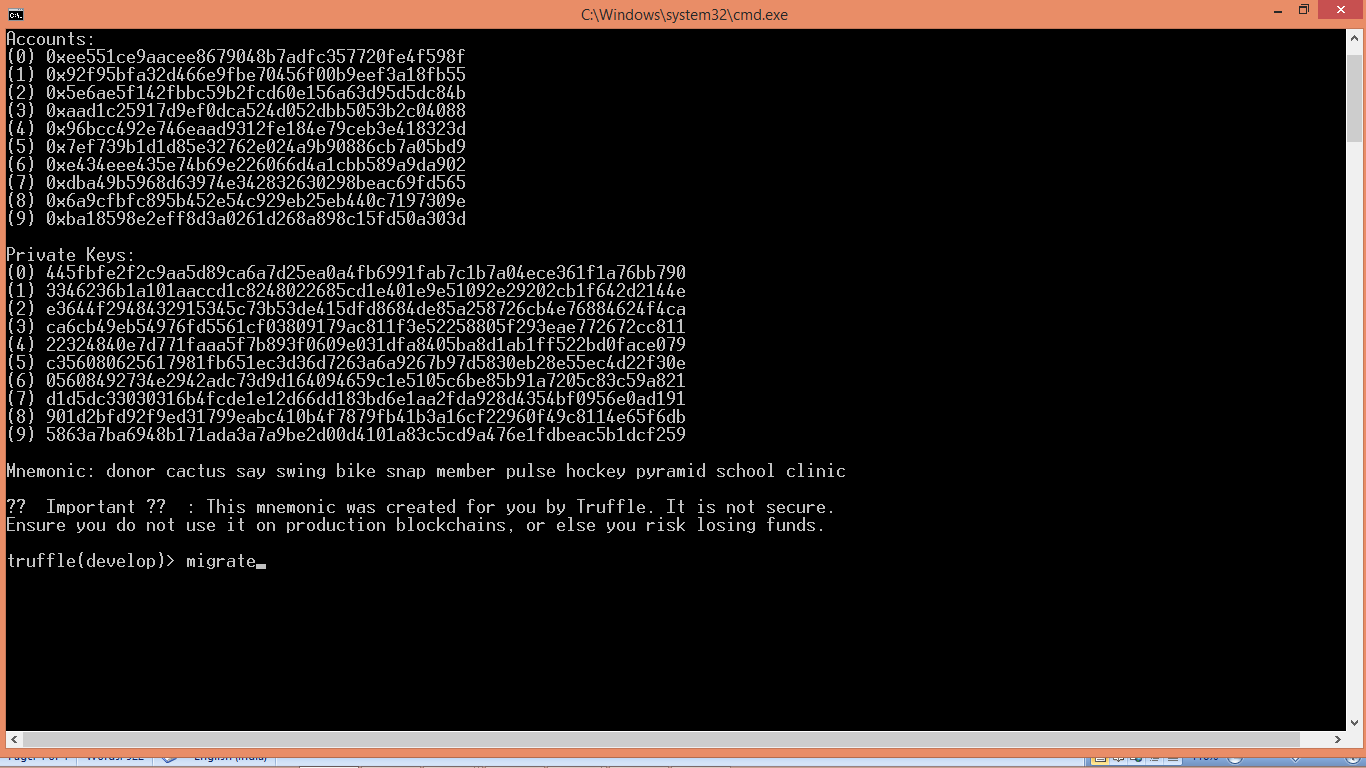
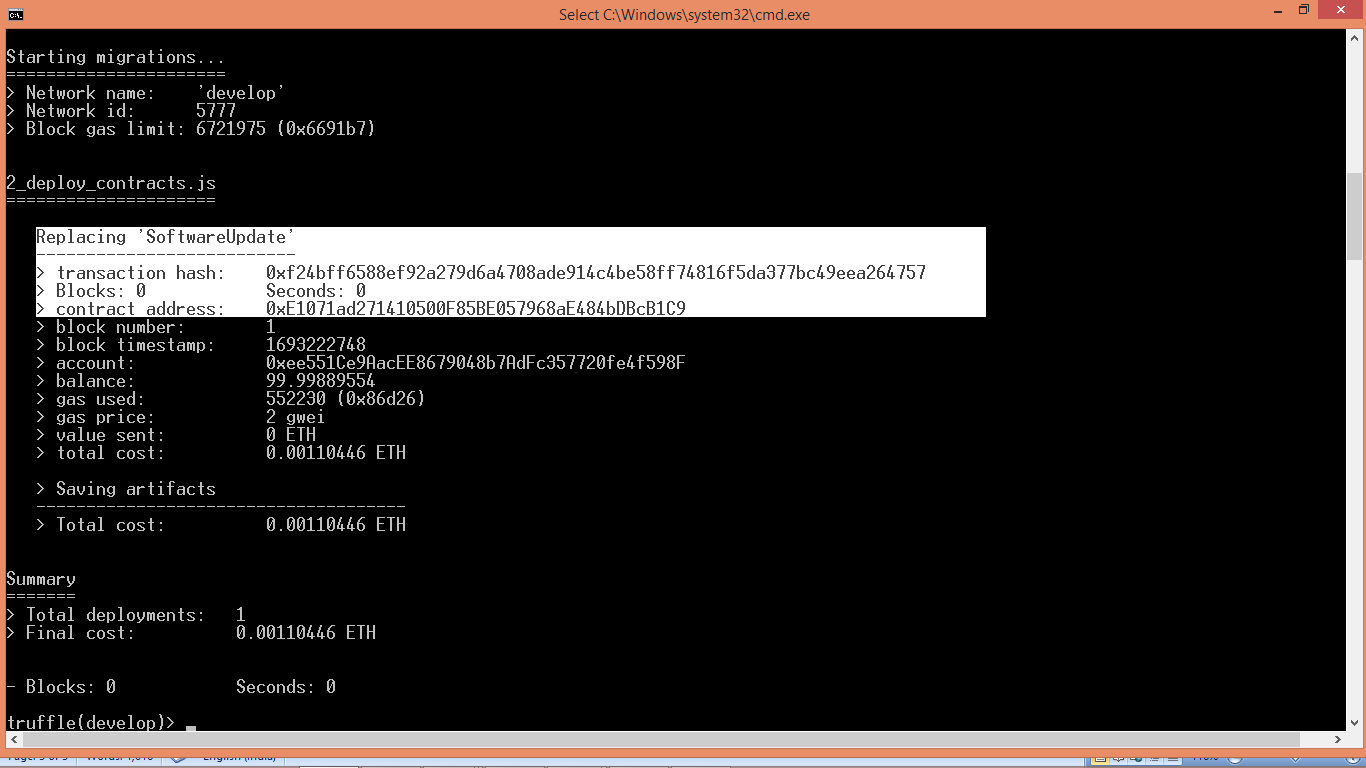
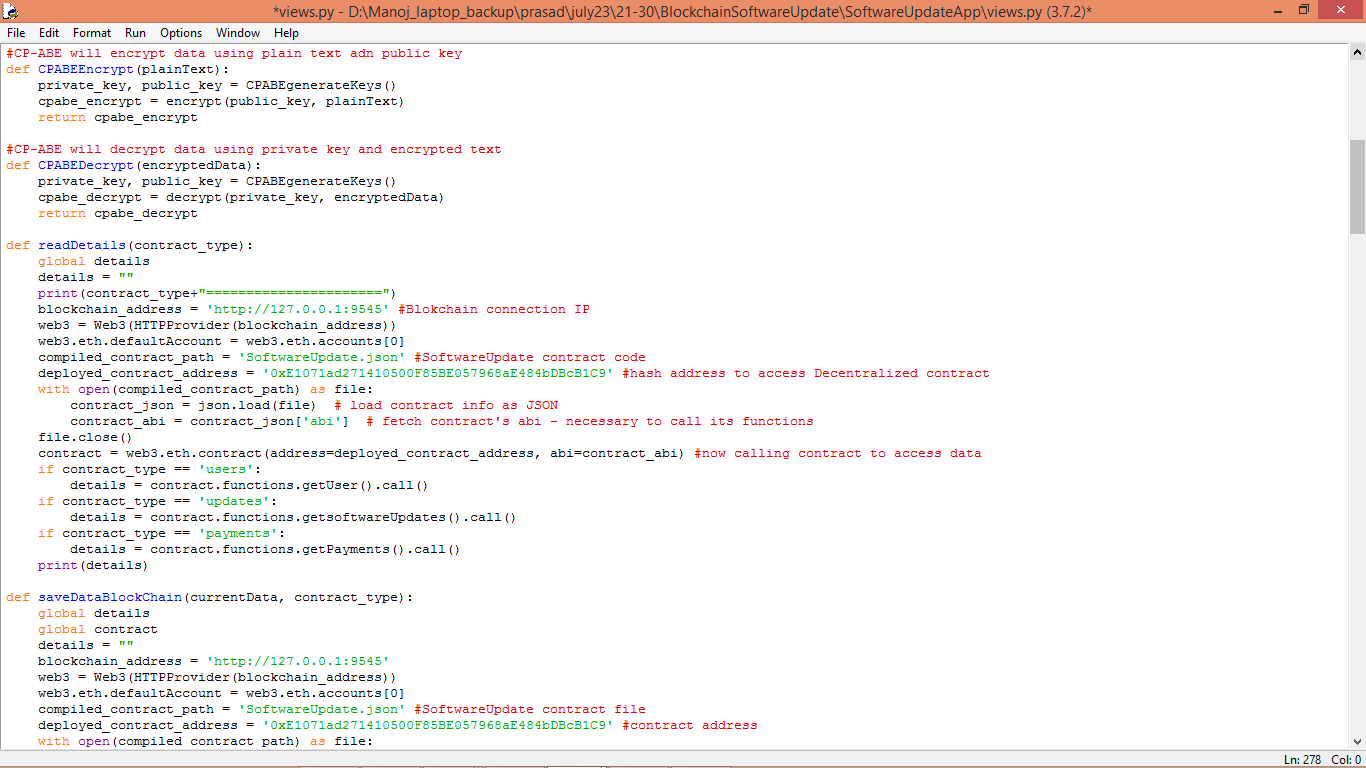
Implementation Details

We don’t have any read IOT devices so we build simulation based IOT application which will inspect Blockchain to download software updates and designed web based application where Manufacturer will upload encrypted software updates and then IOT owner will purchase those updates and make available for IOTS for download.

To store and access data from Blockchain we have designed below smart contract using Solidity programming

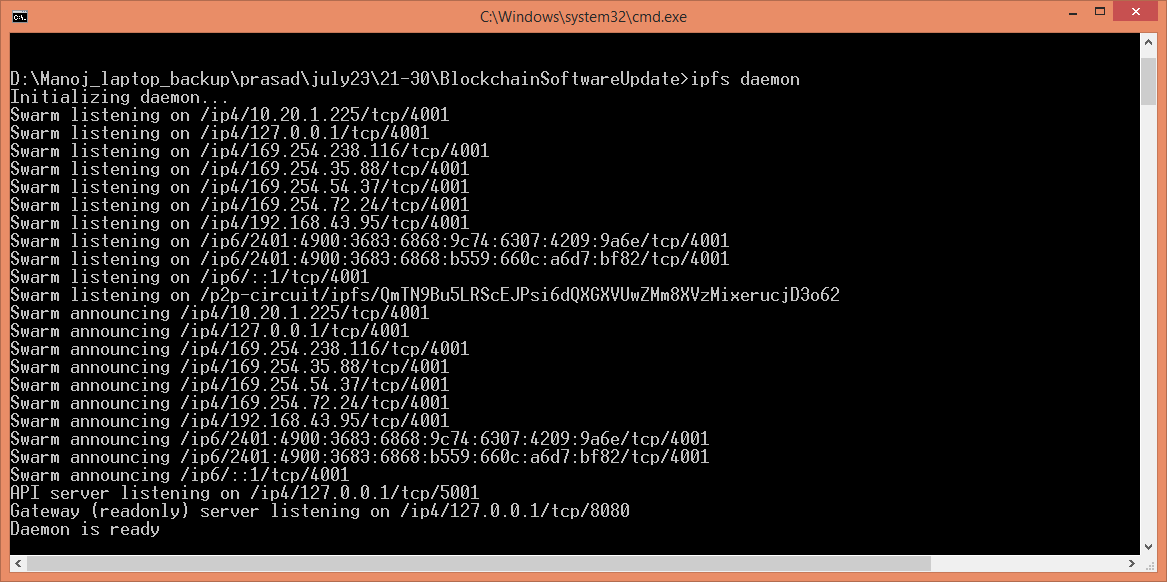


In above contract we have defined function to store and retrieve data from Blockchain and now we need to deployed in Blockchain by using below steps

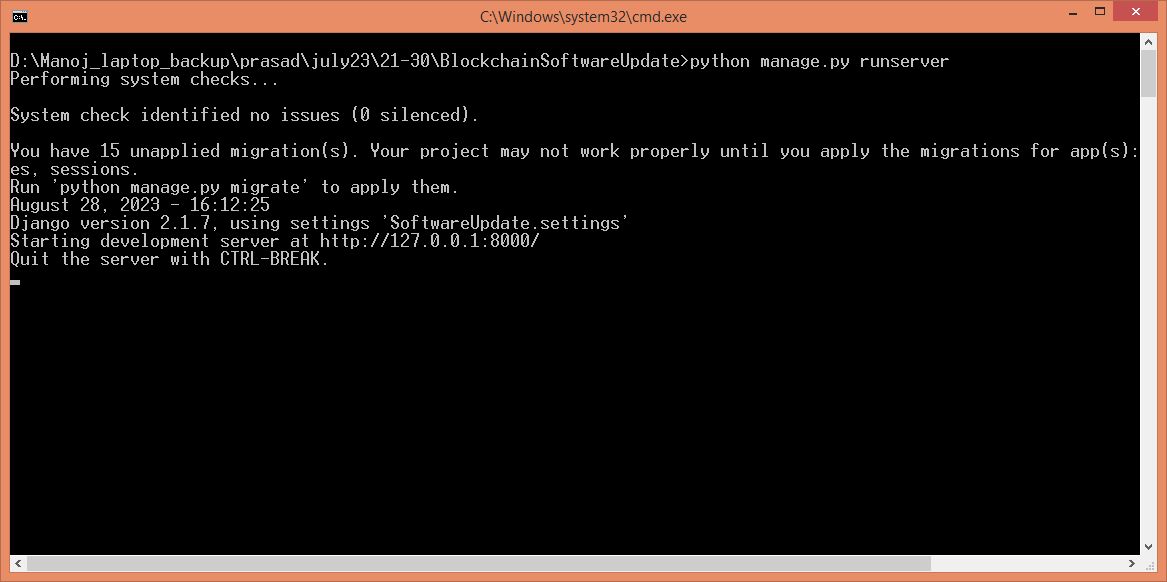
1. Firs go inside ‘hello-eth/node-modules/bin’ folder and then double click on ‘runBlockchain.bat’ file to get below screen
2. 
3. In above screen Blockchain generated default account and private keys and now type command as ‘migrate’ and press enter key to get below page and let the below contract running window running
4. 
5. In above screen in white colour text we can see ‘SoftwareUpdates’ contract deployed and we got contract address also and we need to specify that address in python program to call Smart Contract from python. In below screen showing python code
6. 
7. In above screen read red colour comments to know about contract calling from python

SCREEN SHOTS

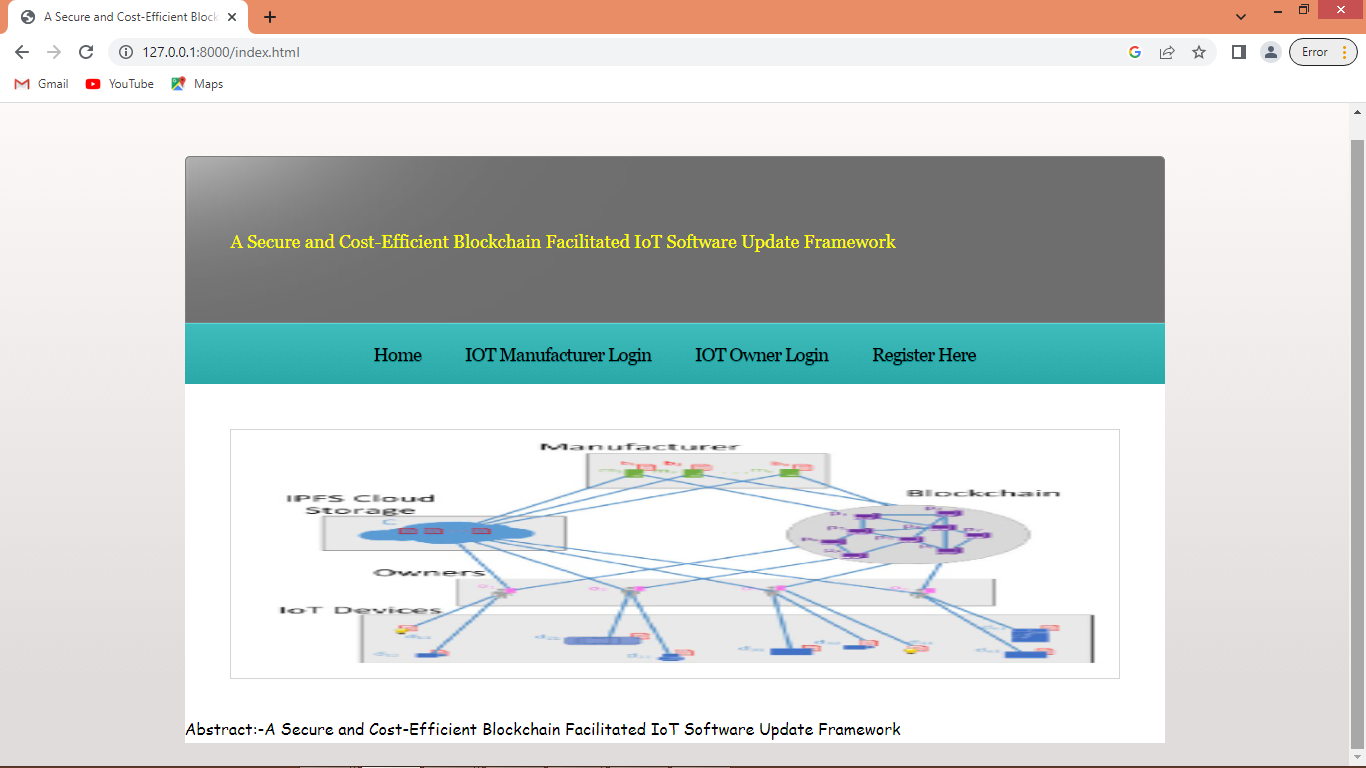
To run project first double click on ‘Start\_IPFS.bat’ file to start IPFS server and get below screen



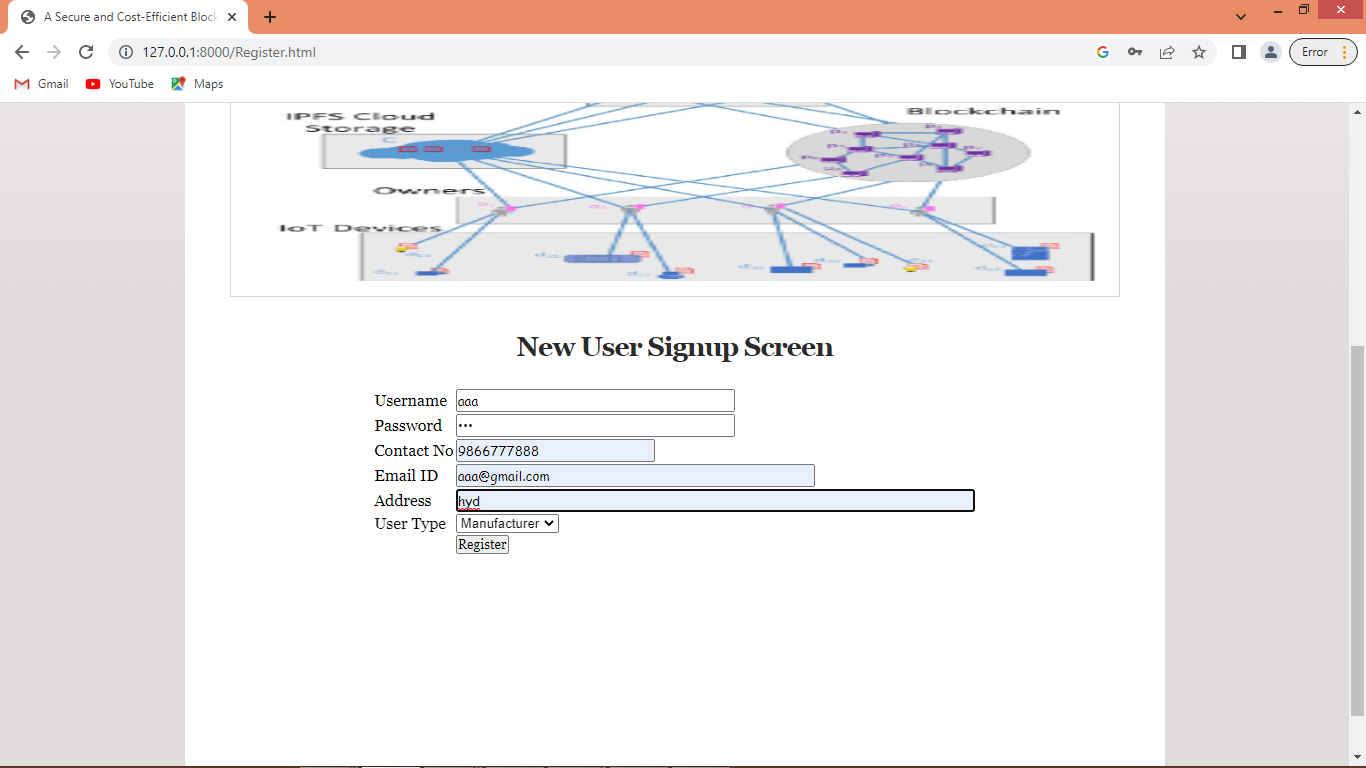
In above screen IPFS server started and now double clicks on ‘run.bat’ file to start python web server and get below page



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 page



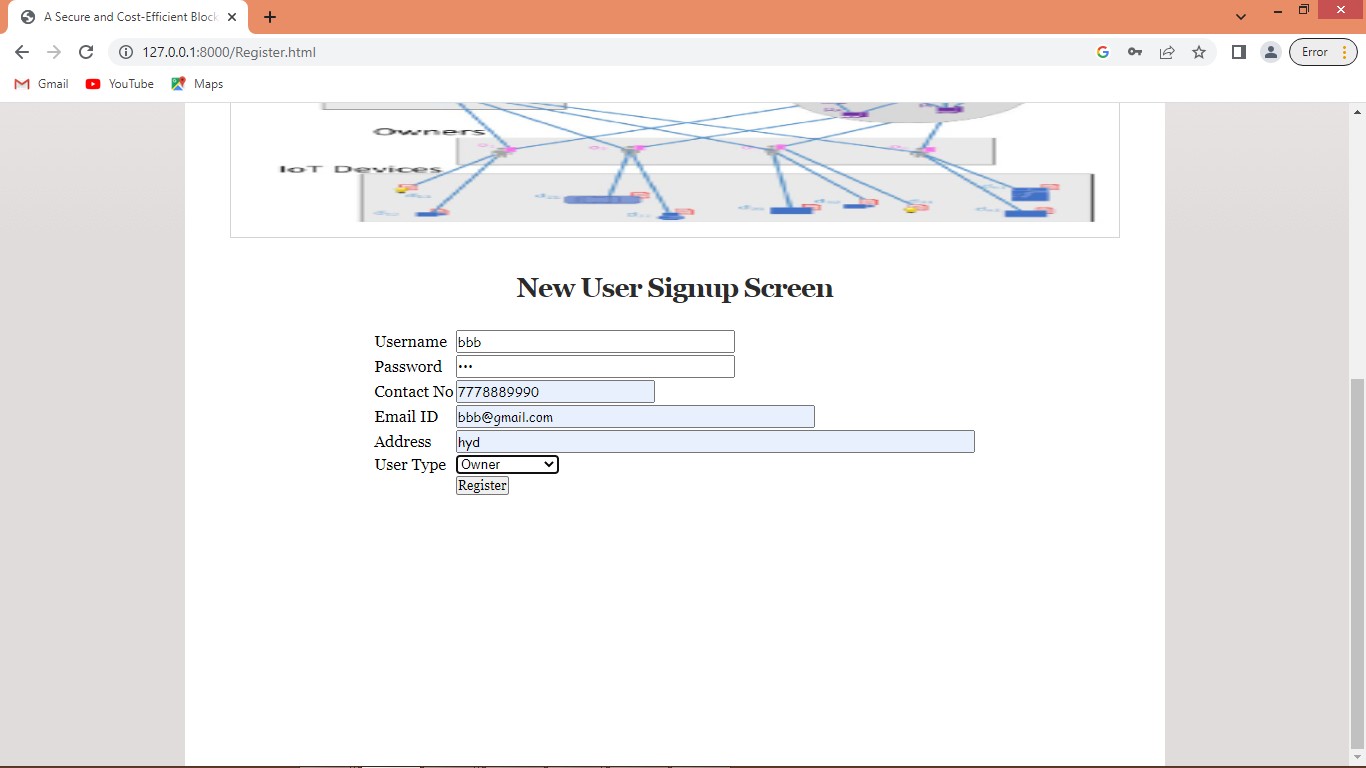
In above screen click on ‘Register Here’ link to get below signup screen and then add manufacturer and owner records



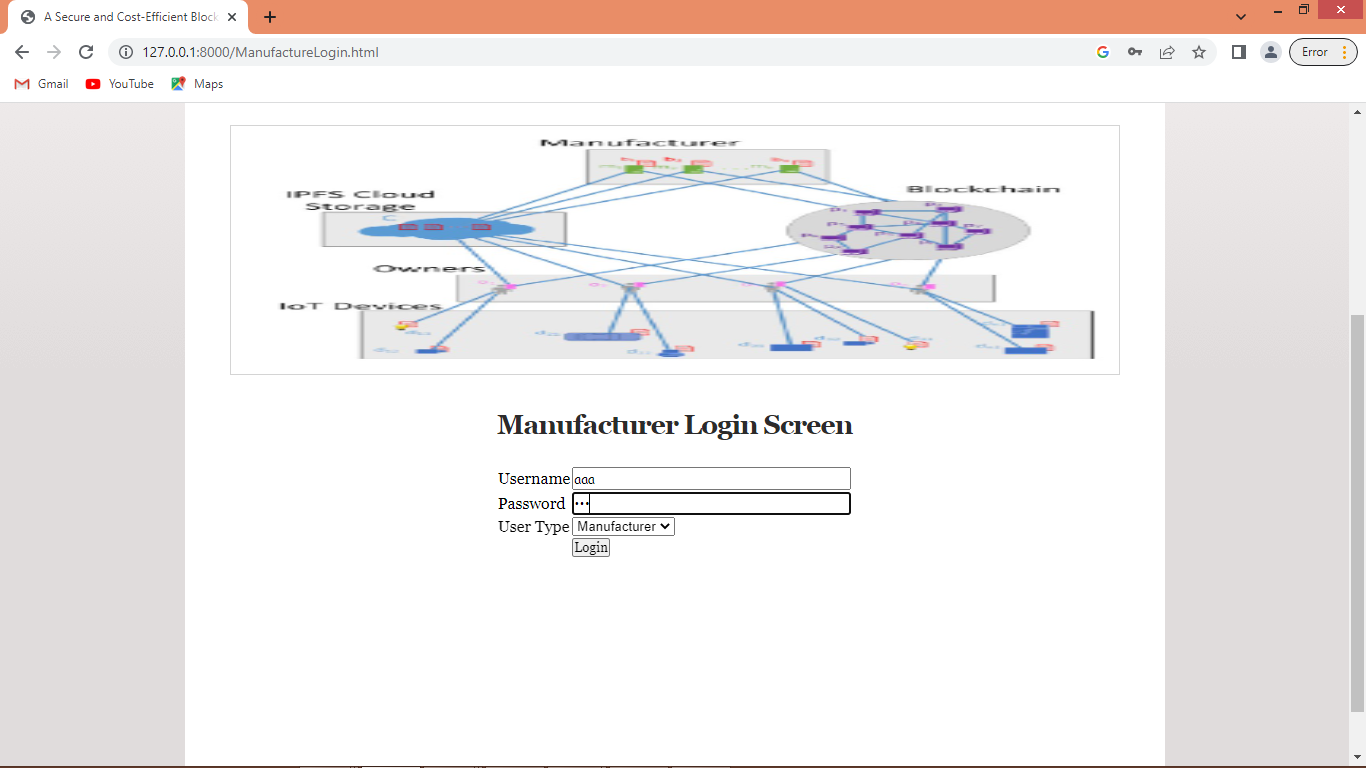
In above screen adding one user as manufacturer and then click ‘Register’ button to add user details to Blockchain and get below page



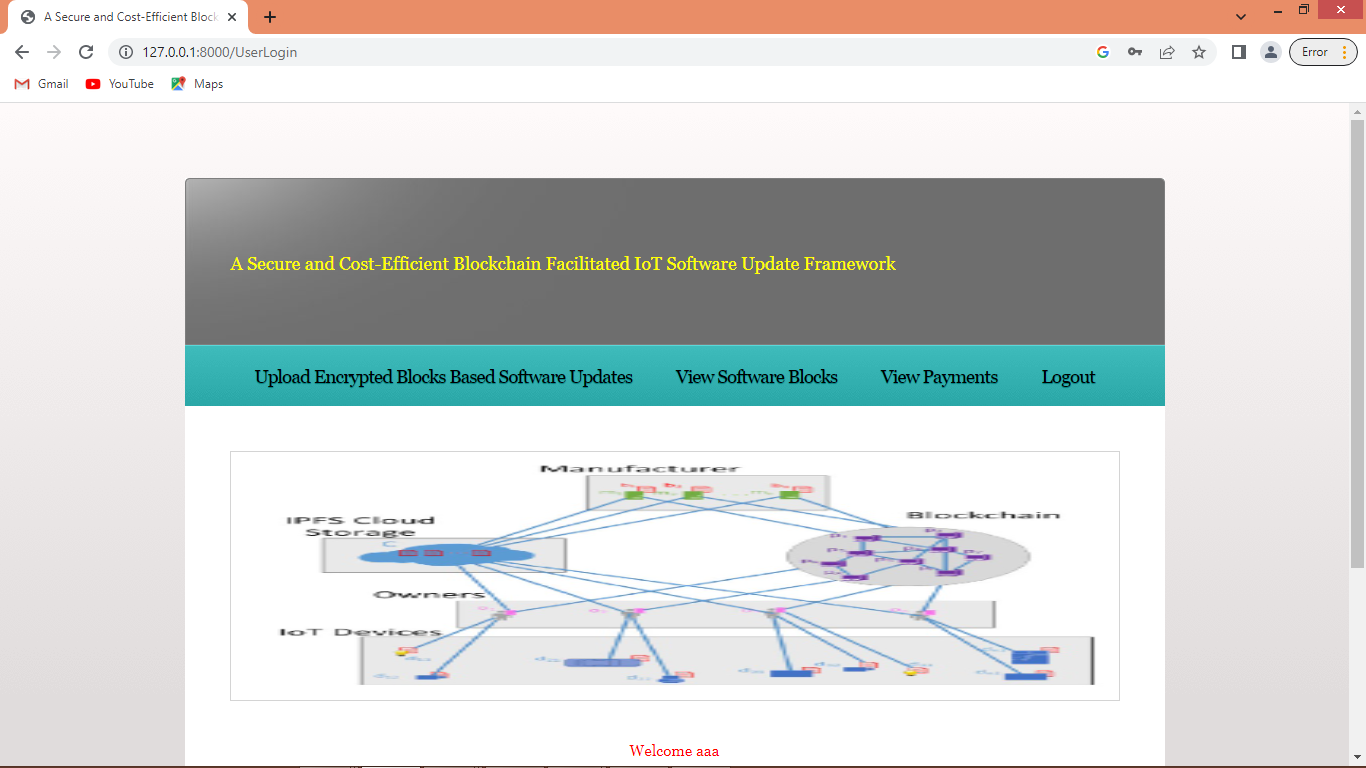
In above screen user details added and now add owner details



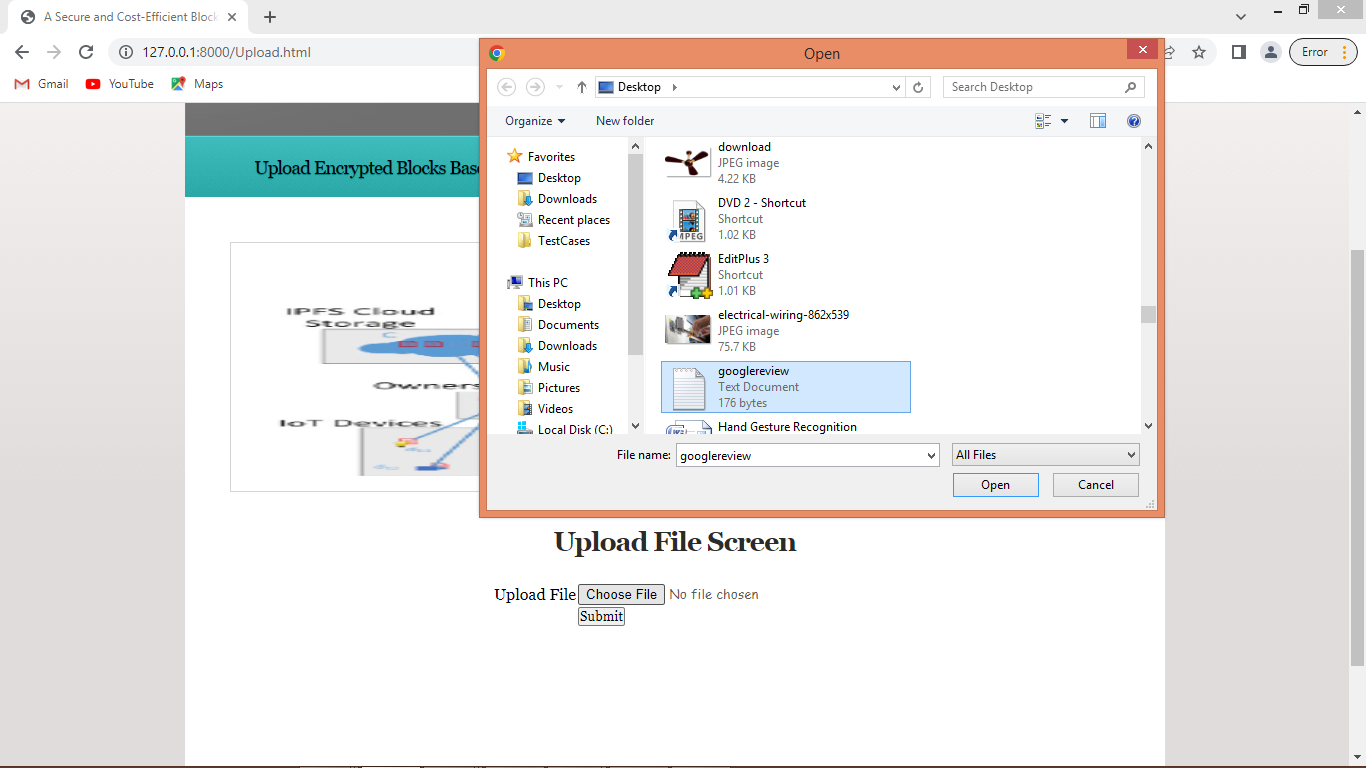
In above screen adding owner details and then press button to save details and then click on ‘Manufacturer Login’ link to get below page



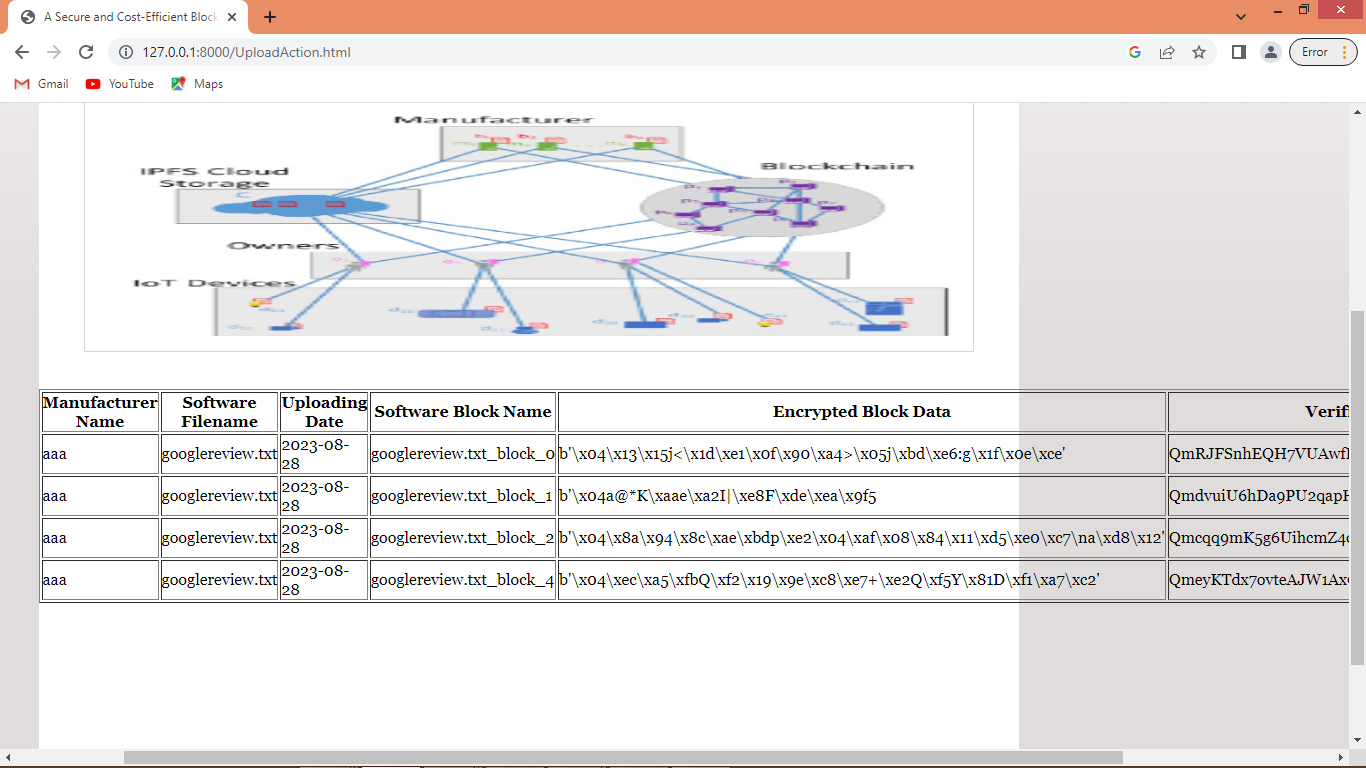
In above screen manufacturer is login and after login will get below page



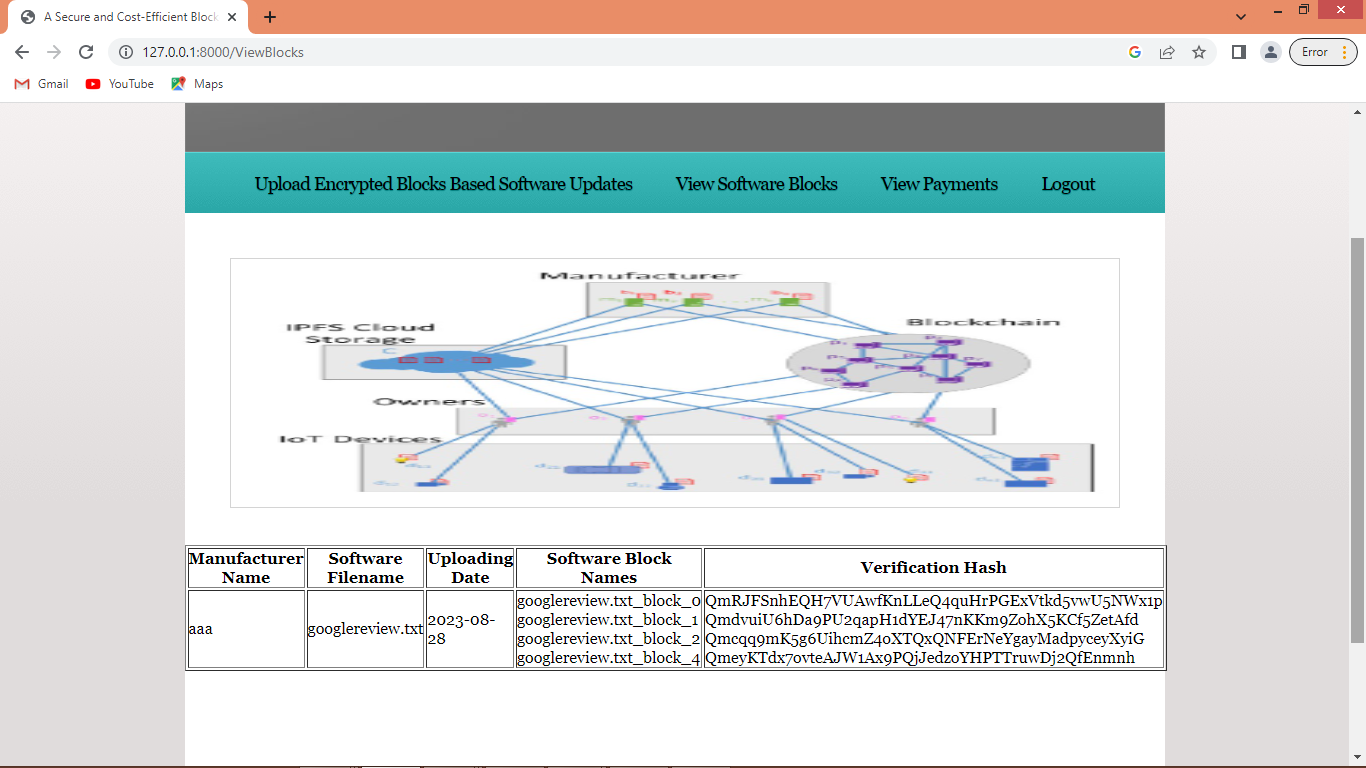
In above screen manufacturer can click on ‘Upload Encrypted Blocks Based Software Updates’ link to upload file and get below page. As upload we don’t have any software so we are uploading file as software updates



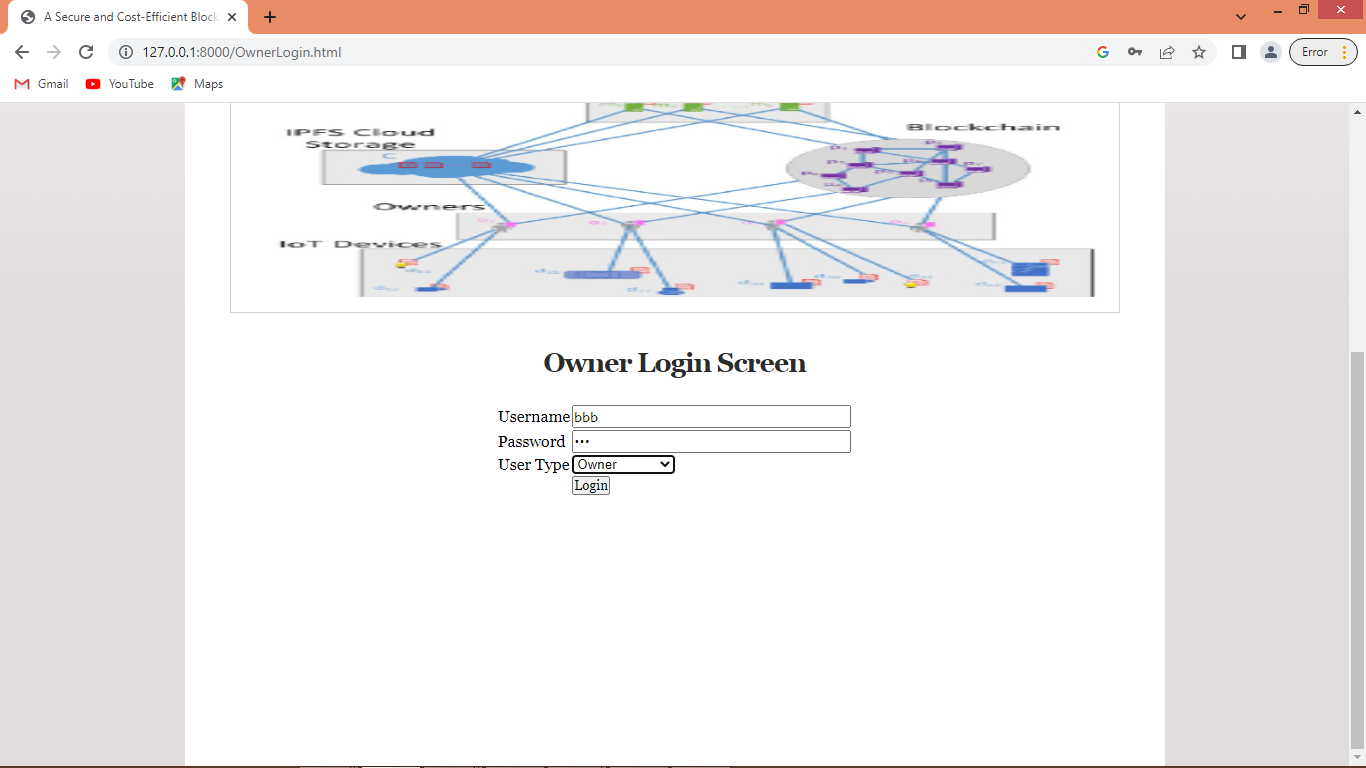
In above screen uploading one file and then click on ‘Open’ button to load file which will divide into multiple blocks and then encrypt and store in IPFS and its address will get store in Blockchain and will get below output



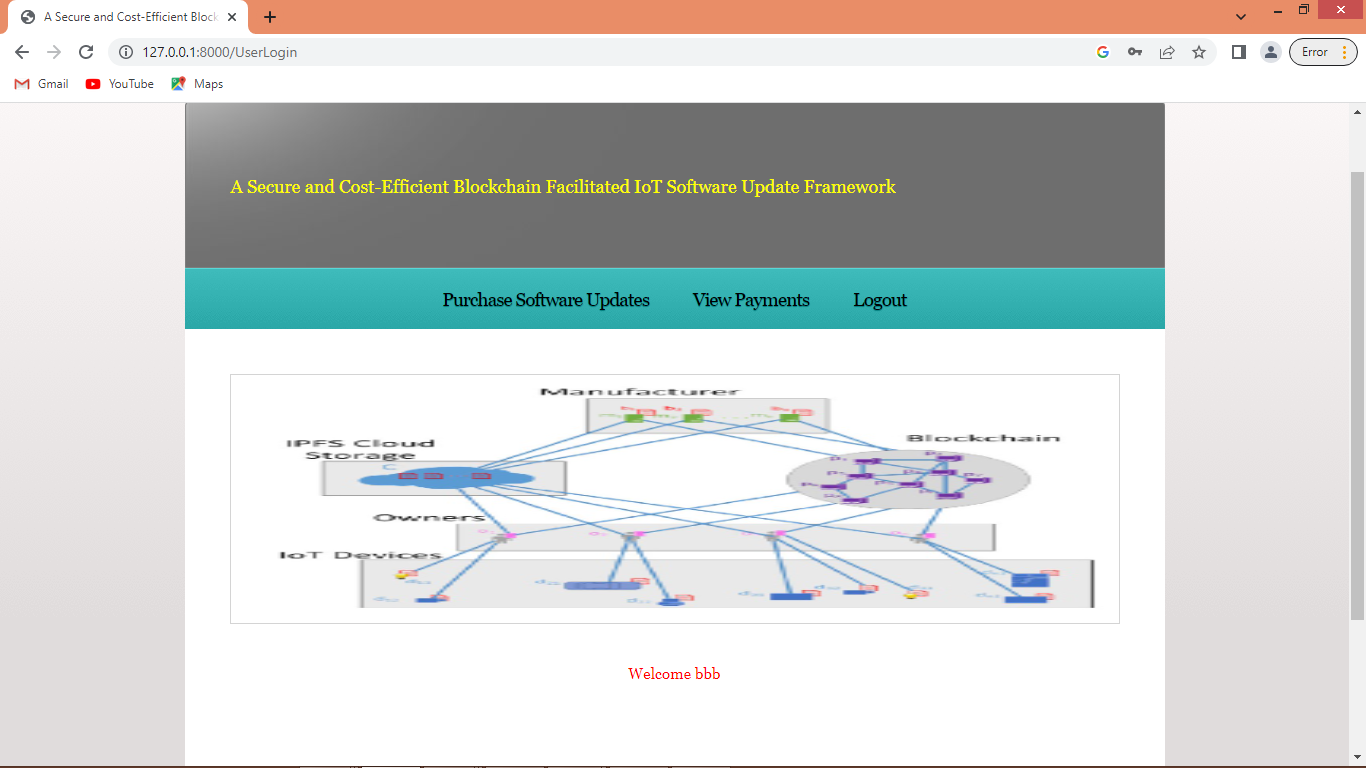
In above screen from same file multiple blocks are generated and we can see name of file, name of blocks, encrypted content and verification code and now can click on ‘View Software Blocks’ link to view blocks of all files like below screen



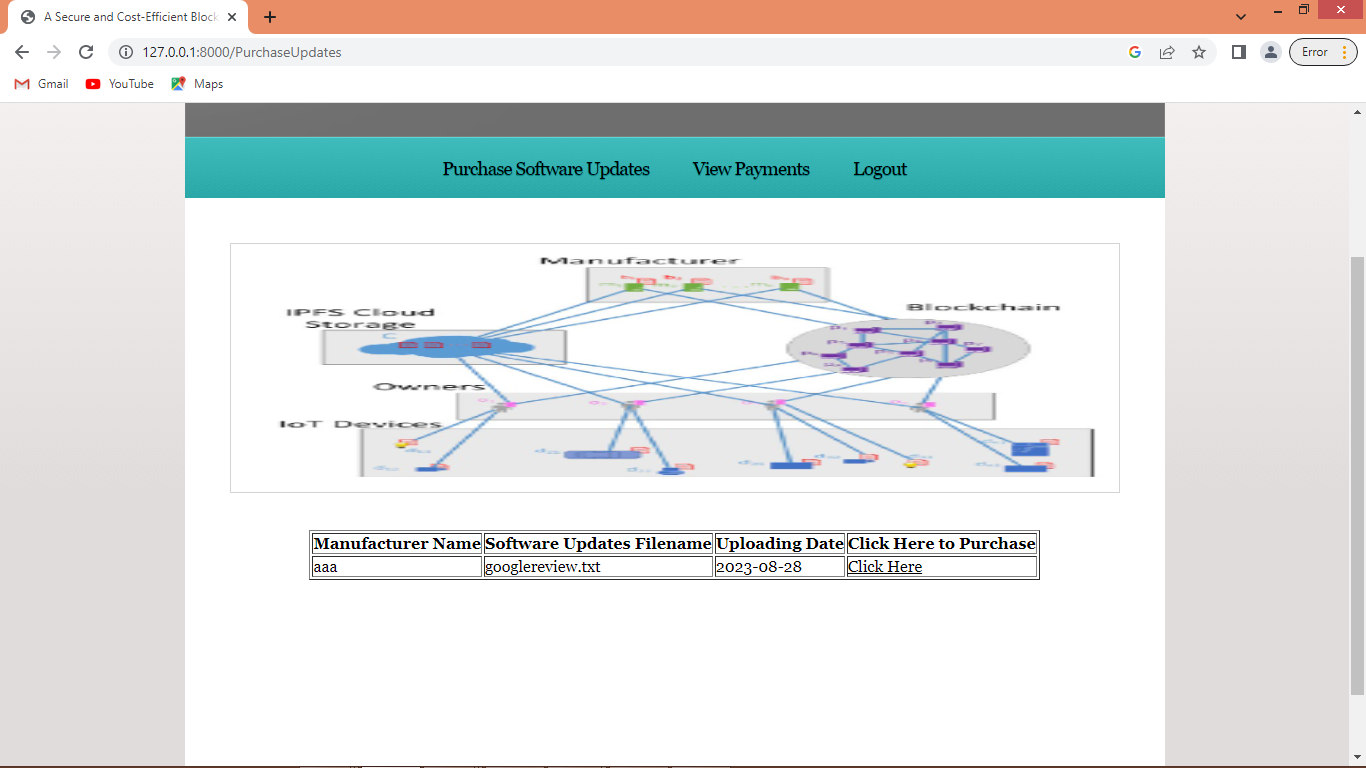
In above screen we can see all files and their blocks and can click on ‘View Payments’ link to view all received payments from owners and now logout and login as IOT owner to purchase uploaded files software updates



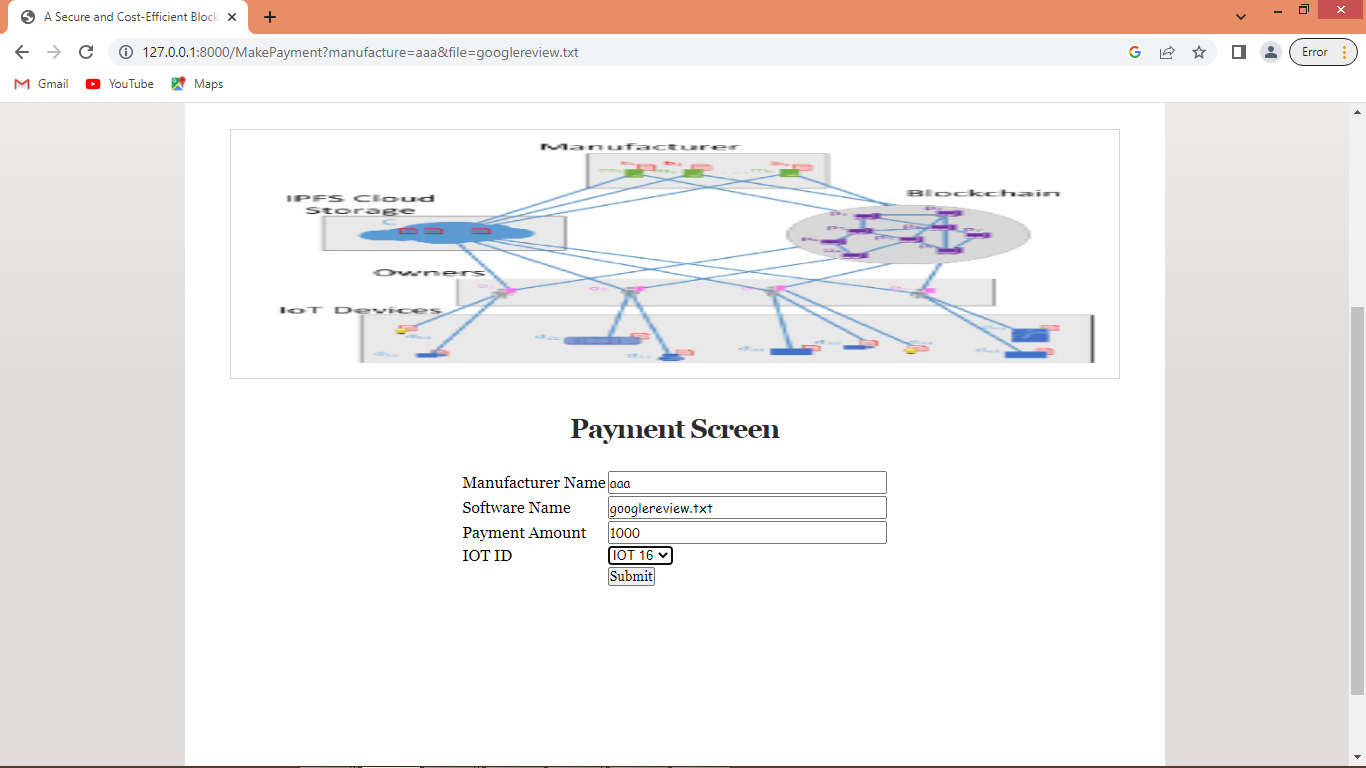
In above screen owner is login and after login will get below page



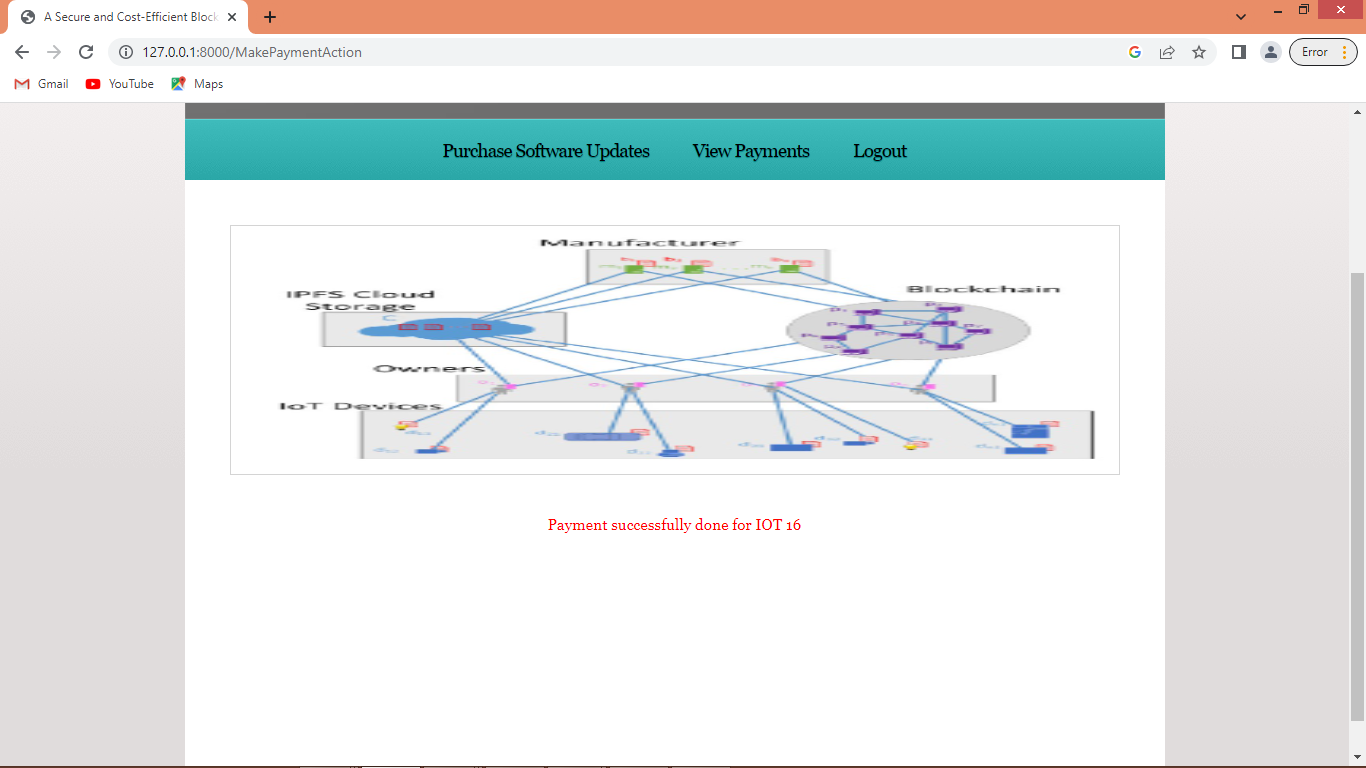
In above screen owner can click on ‘Purchase Software Updates’ link to view all available updates and can purchase desired updates



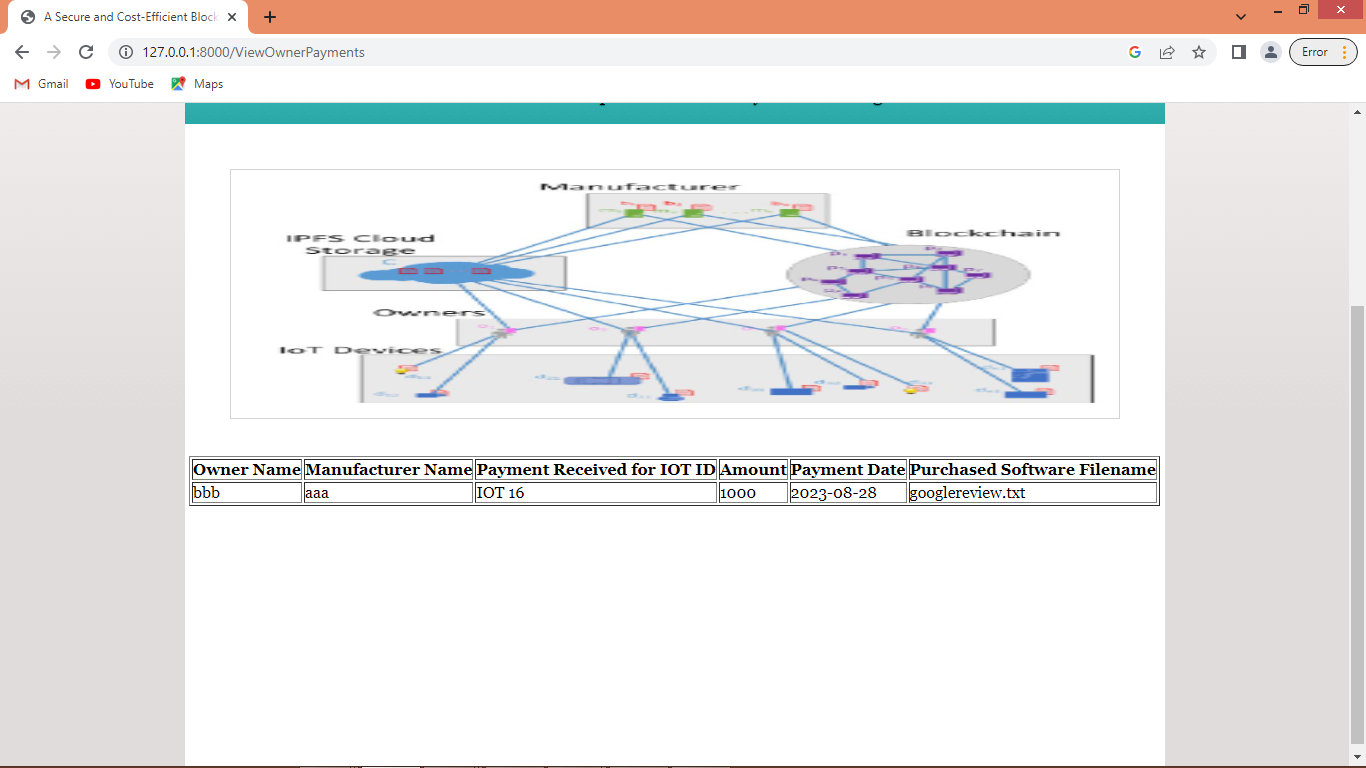
In above screen owner can view list of updates and then click on ‘Click Here’ link to purchase and get below page



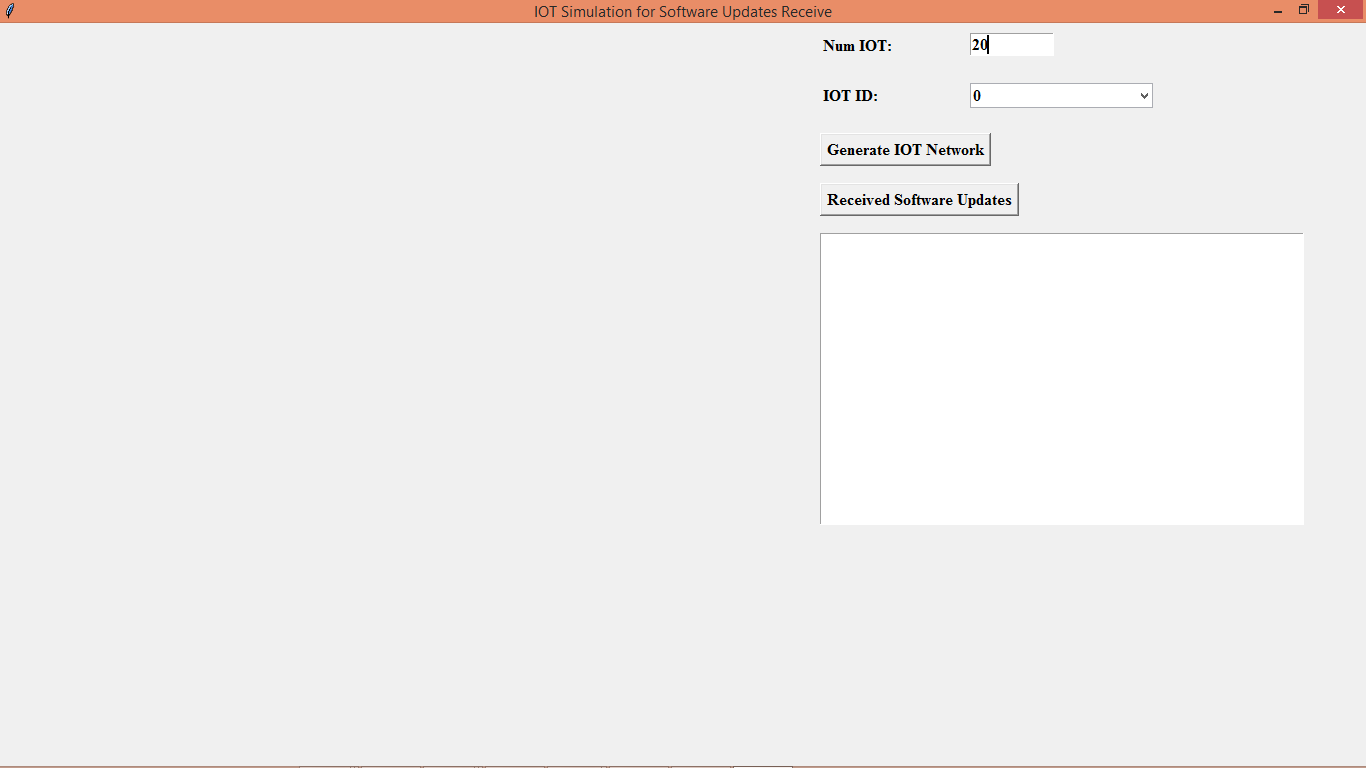
In above screen IOT purchasing software update for IOT16 and in simulation output we can see IOT16 will be receiving and updating updates. Now click on ‘Submit’ button to complete payment and get below page



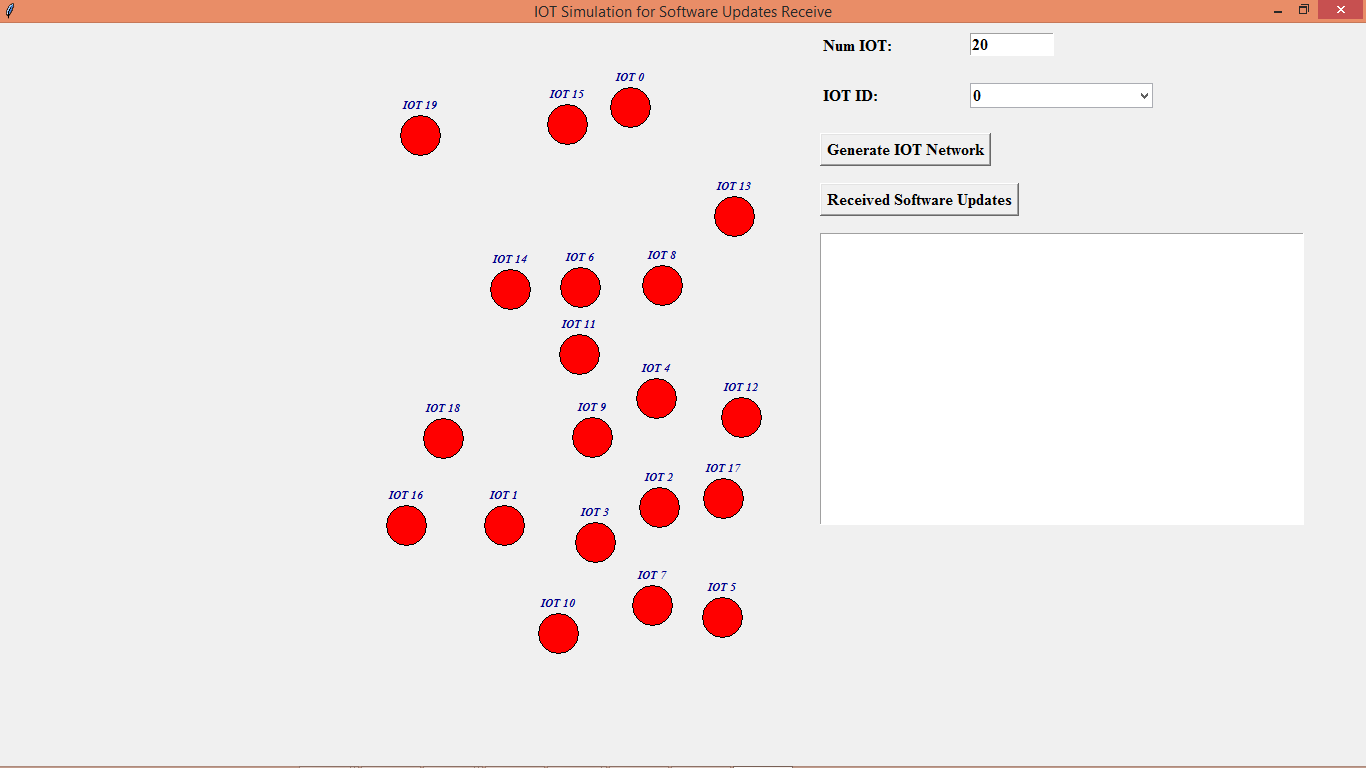
In above screen in red colour text we can see payment completed and now click on ‘View Payments’ link to view all payments done by this owner and get below page



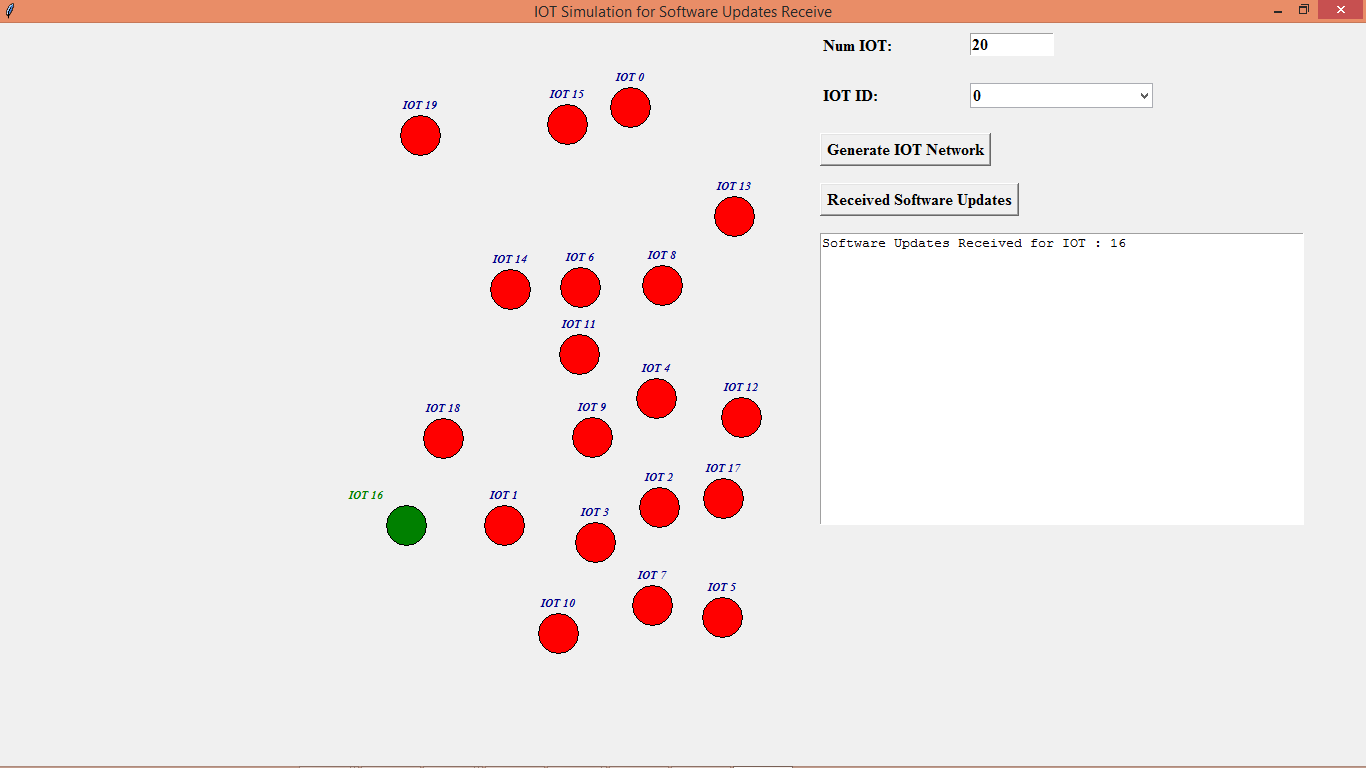
In above screen owner can see list of software updates purchased and now double click on ‘runIOTSimulation.bat’ file to get below simulation file



In above screen enter ‘Num IOT ‘ as 20 and press ‘Generate IOT Network’ to get below screen



In above screen all red colour circles are consider as IOT nodes and are placed at different location and now click on ‘Received Software Updates’ button to download and update software only for those IOT’s purchased by owner



In above screen we can see the IOT for which IOT owner purchase updates will receive and changed its colour to green to indicate as its receiving updates.