BLOCKCHAIN PROJECT(BITS-F452)
REPORT- CHARITY TRACKER

-USNEEK SINGH, 2019A7PS0127P
-AKASH S REVANKAR, 2019A7PS0294P

Problem Statement

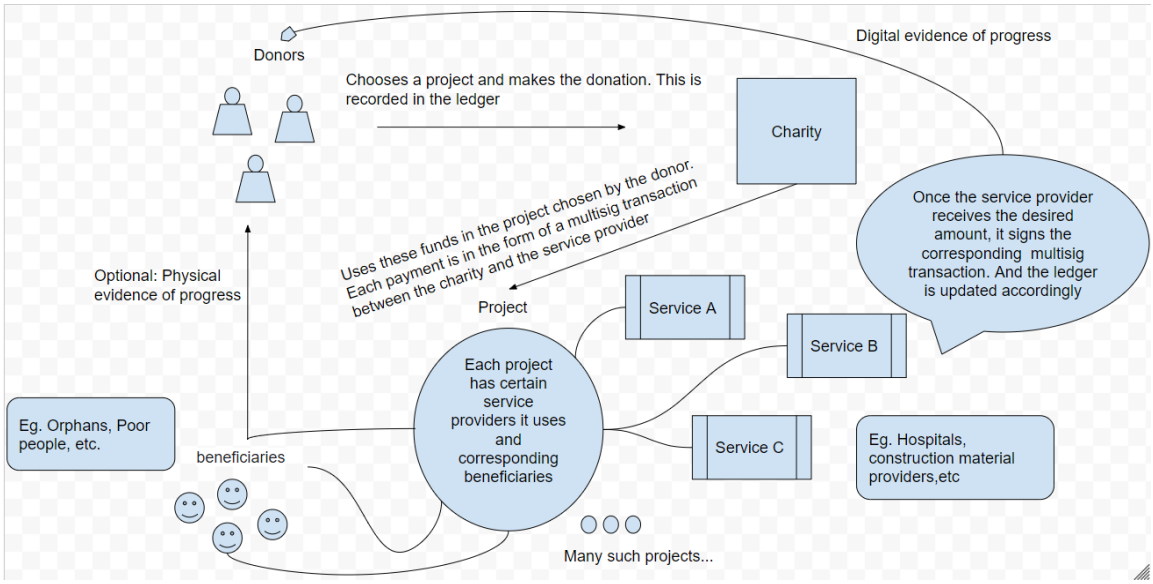
It has been observed that people feel insecure about making donations to charity organisations on account of their money being misused. Every year, tens of millions of dollars are misspent by charities and the general public is losing trust. It was found that the cash misspent by charities over the past decade amounted to nearly a billion dollars. People would feel more secure towards a platform that is transparent and effective. NGOs and other charity organizations can increase trust among people if they provide them with information on how and where their donation is being used. The most effective way to track the spending of donation in a trustworthy manner is by using Blockchain Technology.

Assumptions and Objectives

Objectives

This project aims to strengthen the philanthropic impact by facilitating a more secure process for collecting and spending funds all over the world. This project implements Blockchain Technology to make donations transparent by providing real-time updates of donation outcomes and thus increasing the trust of donors and philanthropists in charity organisations.
This projects aims to work in following manner:-

1. Donor will choose a project and make the donation. This is recorded in the ledger.
2. Charity would use these funds in the project chosen by the donor. Each payment is in the form of a multisig transaction between the charity and the service provider.
3. Each project has certain service providers it uses and corresponding beneficiaries
4. Once the service provider receives the desired amount, it signs the corresponding multisig transaction. And the ledger is updated accordingly. This would be the digital evidence of progress.



Assumptions

Due to various constraints all the objectives could not be implemented as expected and we had to make some assumptions which are stated as follows:-

1. Donors should donate money in ether.
2. Service providers should accept money in ether.

Description and concepts required

1. **Hyperledger Besu**
   Hyperledger Besu is an Ethereum client designed to be enterprise-friendly for both public and private permissioned network use cases. It can also be run on test networks such as Rinkeby, Ropsten,etc. An Ethereum clients contains:

   - An execution environment for processing transactions in the Ethereum blockchain
   - Storage for persisting data related to transaction execution
   - Peer-to-peer (P2P) networking for communicating with the other Ethereum nodes on the network to synchronize state
   - APIs for application developers to interact with the blockchain

   Hyperledger Besu includes several consensus algorithms including PoW, and PoA (IBFT, IBFT 2.0, Etherhash, and Clique). In our project we have used the **clique protocol.**
   Clique is more fault-tolerant than traditional consensus algorithms (IBFT 2.0). Clique tolerates up to half of the validators failing. IBFT 2.0 networks require greater than or equal to ⅔ of validators to be operating to create blocks.

2. **Alethio Ethereum Lite Explorer**
   The Alethio Ethereum Lite Explorer is a Web application that connects to any Ethereum JSON-RPC-enabled node. The Explorer does not require an online server, hosting, or trusting third parties to display the blockchain data. It presents the data of our private network containing information about the node, its address and current state in the blockchain.

3. **Private Network On Metamask**
   We can set up our own private network on metamask following these steps:-

   1. After you sign in to MetaMask, connect to the private network RPC endpoint:
   2. In the MetaMask network list, select Custom RPC.
   3. In the New RPC URL field, enter the JSON-RPC HTTP service endpoint displayed when you started the private network..
   4. Make the transactions from your account to other accounts on this private network

Snapshots of working of the project

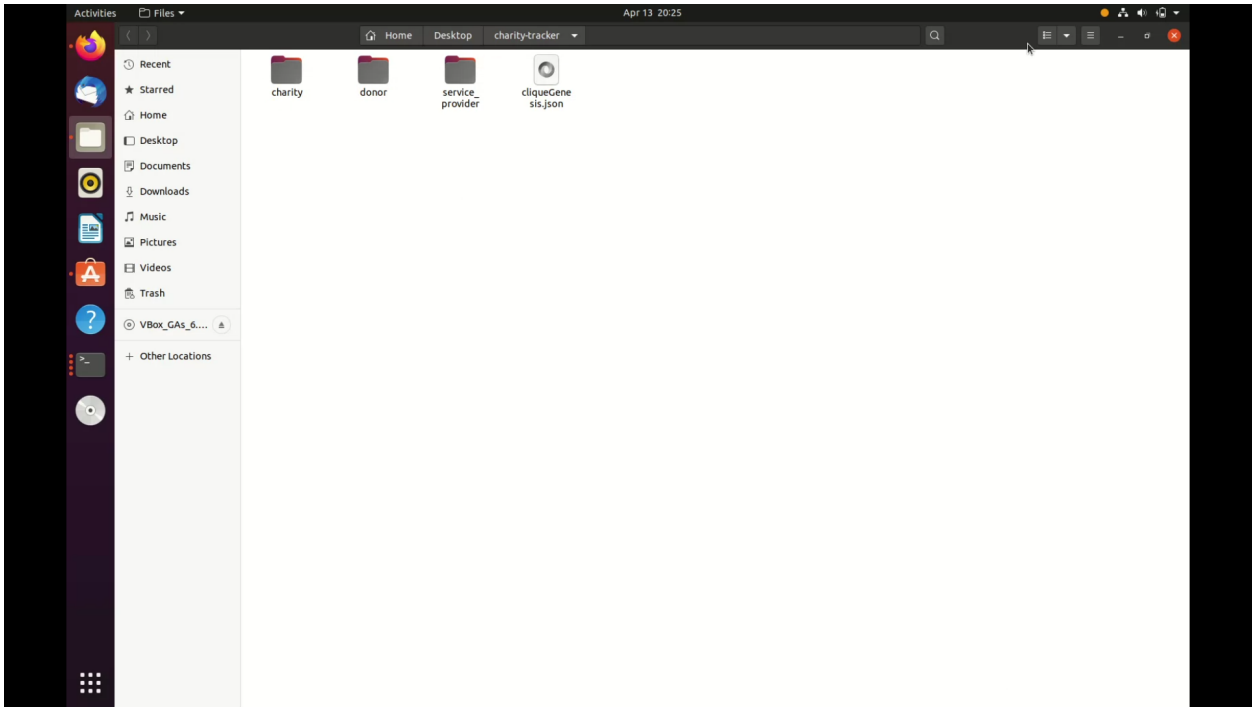


Fig 1. Three nodes created: charity, donor and service_provider and genesis file for the clique private network added

Fig 2. All three nodes added to the allowed-list in permissions_config.toml

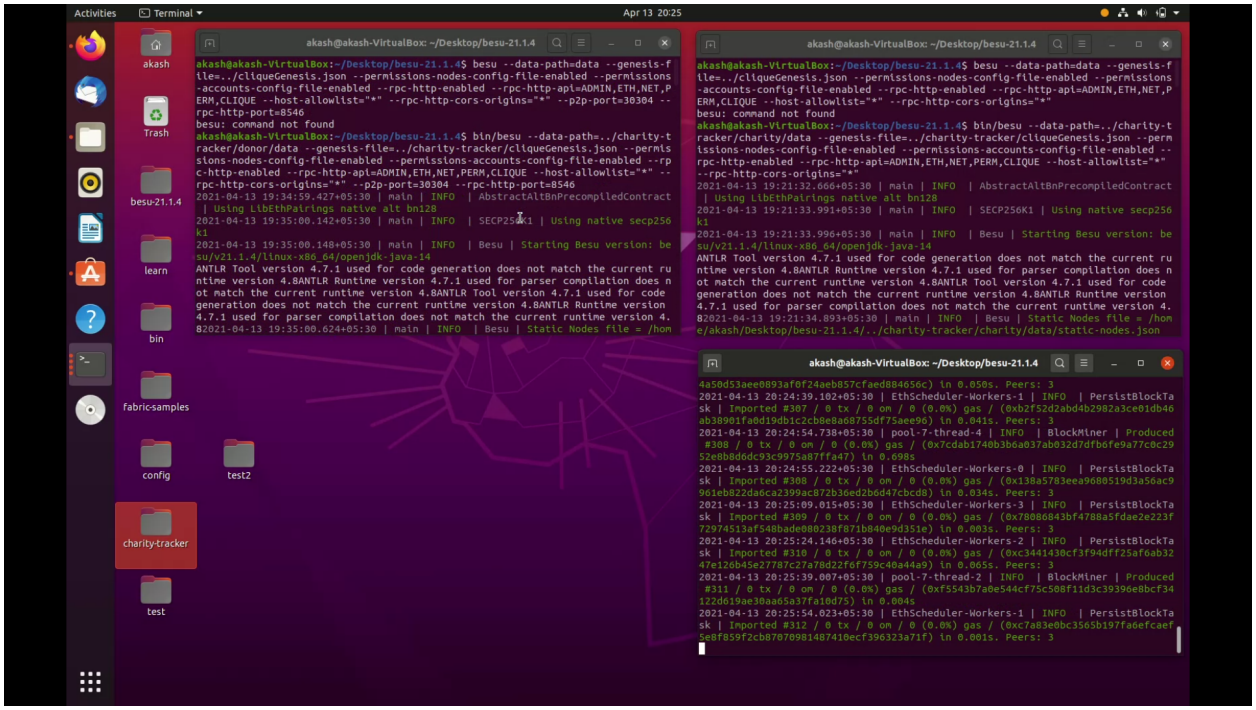


Fig 3. All three nodes started with --permissions-accounts-config-file-enabled

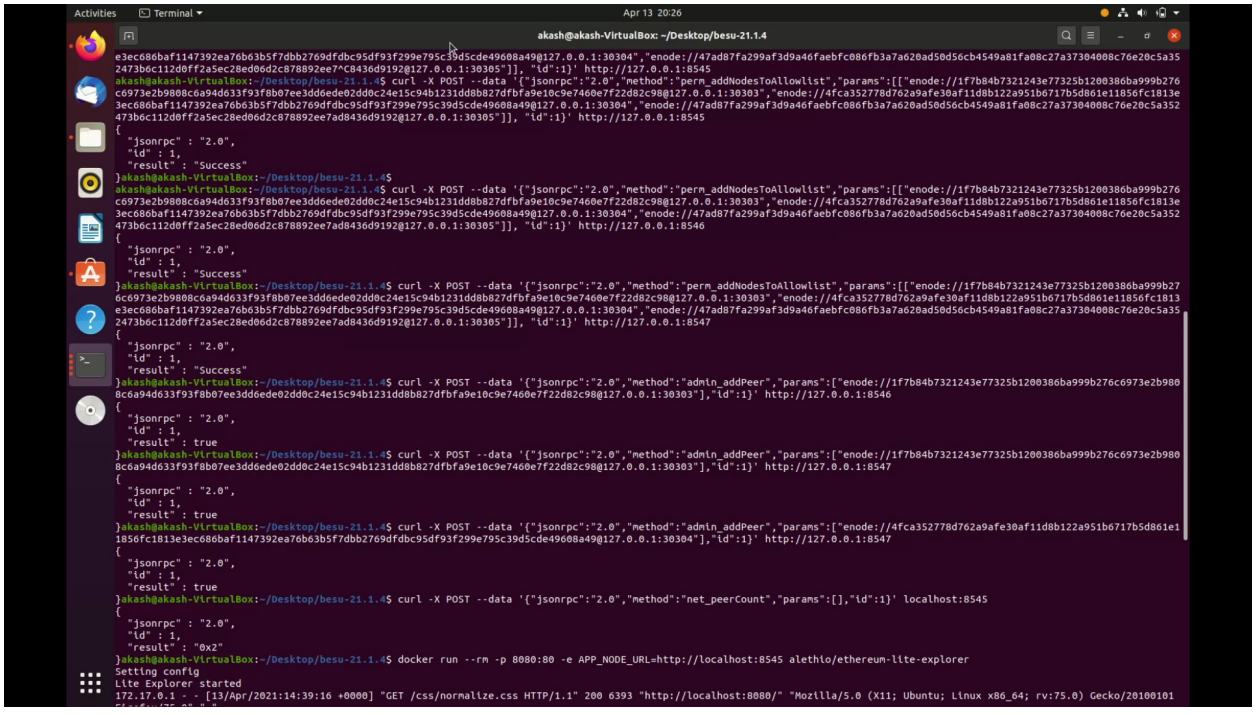


Fig 4. Used the perm_addNodesToAllowlist JSON-RPC API method to add the nodes to the permissions configuration file for each node. Used net_peerCount JSON-RPC API method to confirm that 'charity' node has two peers (donor and service_provider)

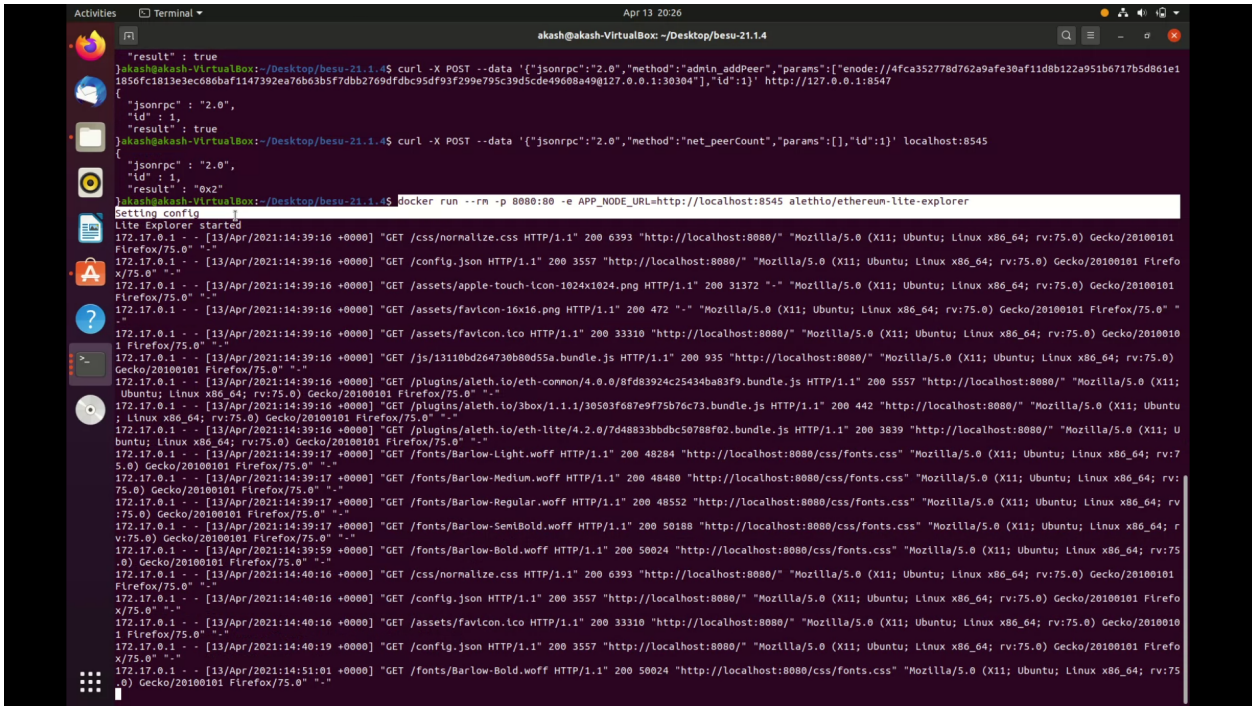


Fig 5. Block-explorer started for our private permissioned network

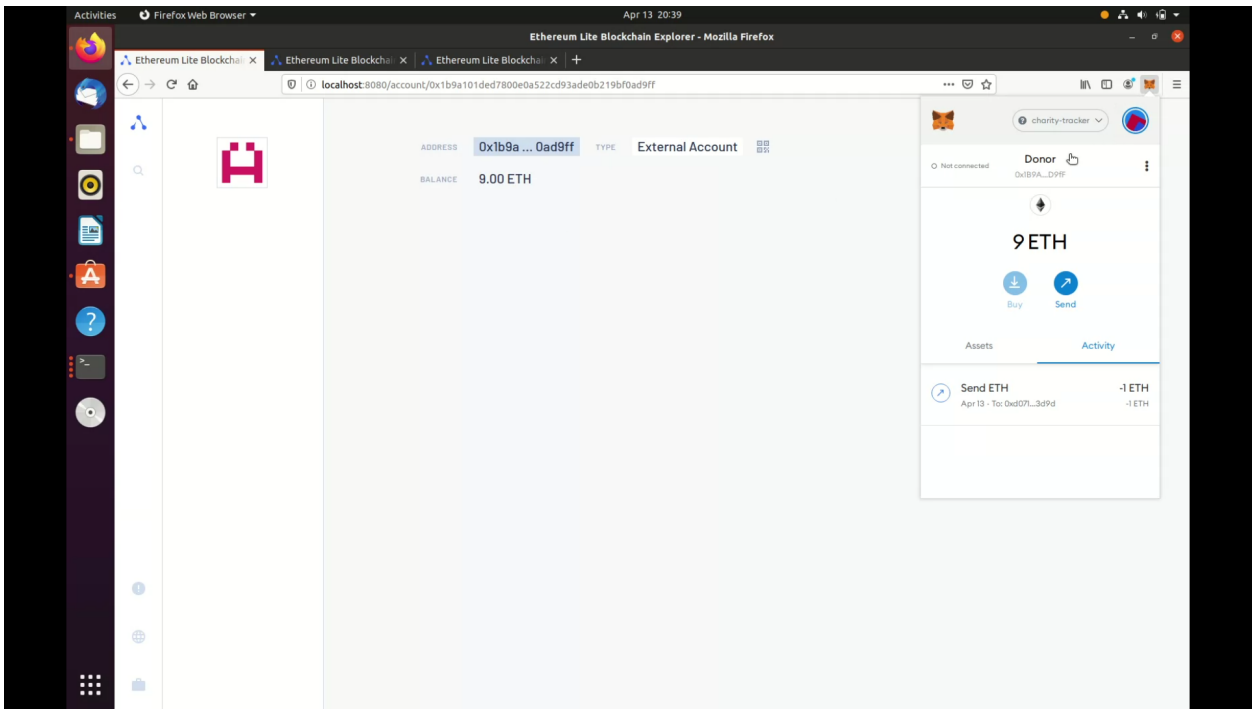


Fig 6. Donor node with 9 ETH balance

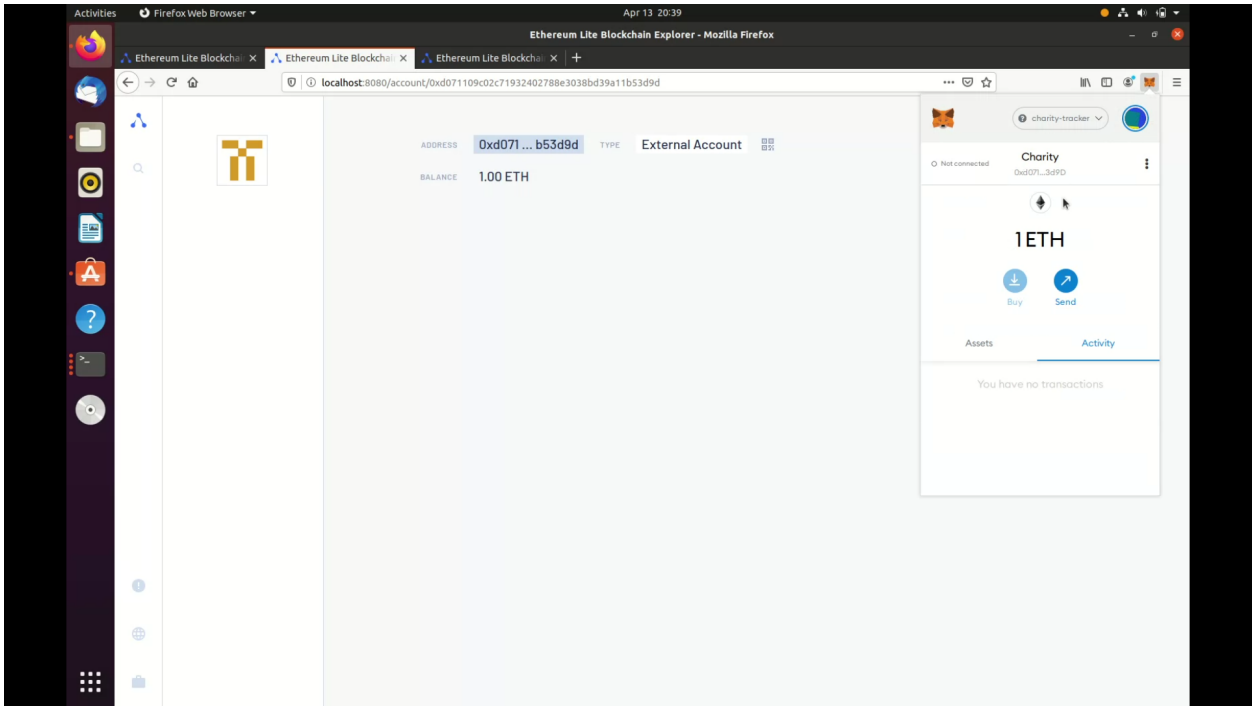


Fig 7. Charity node with 1 ETH balance

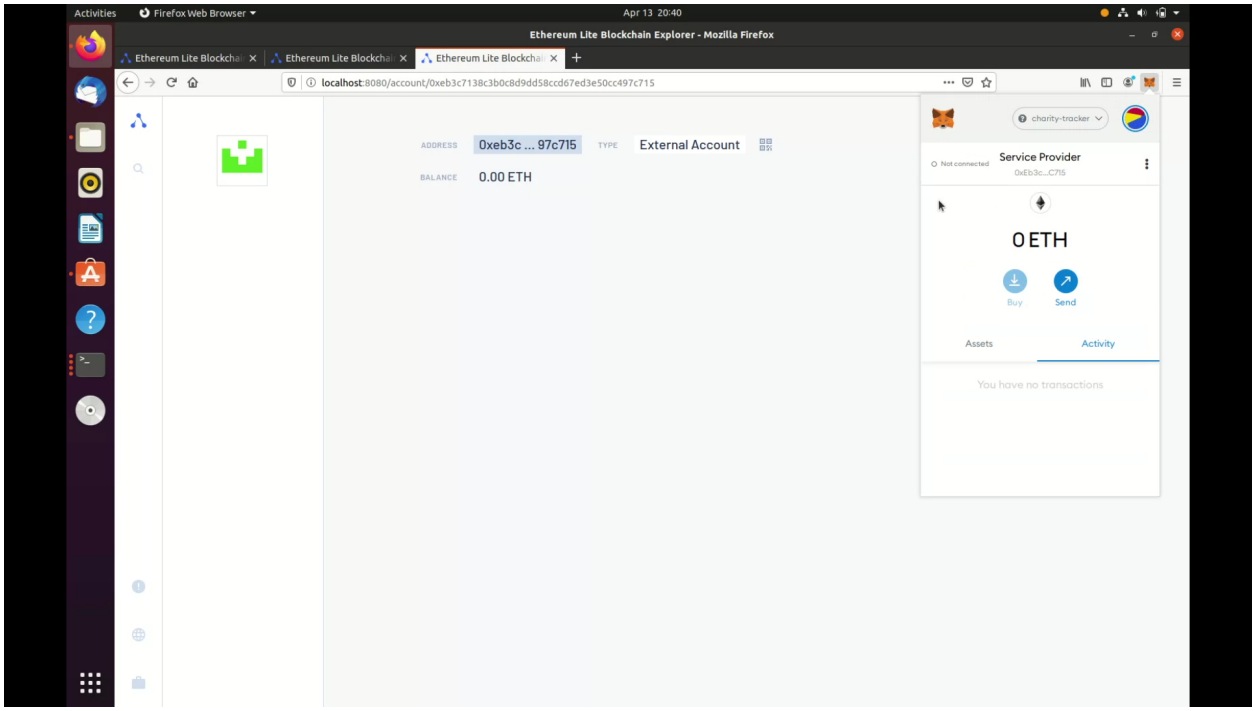


Fig 8. Service Provider node with 0 ETH balance

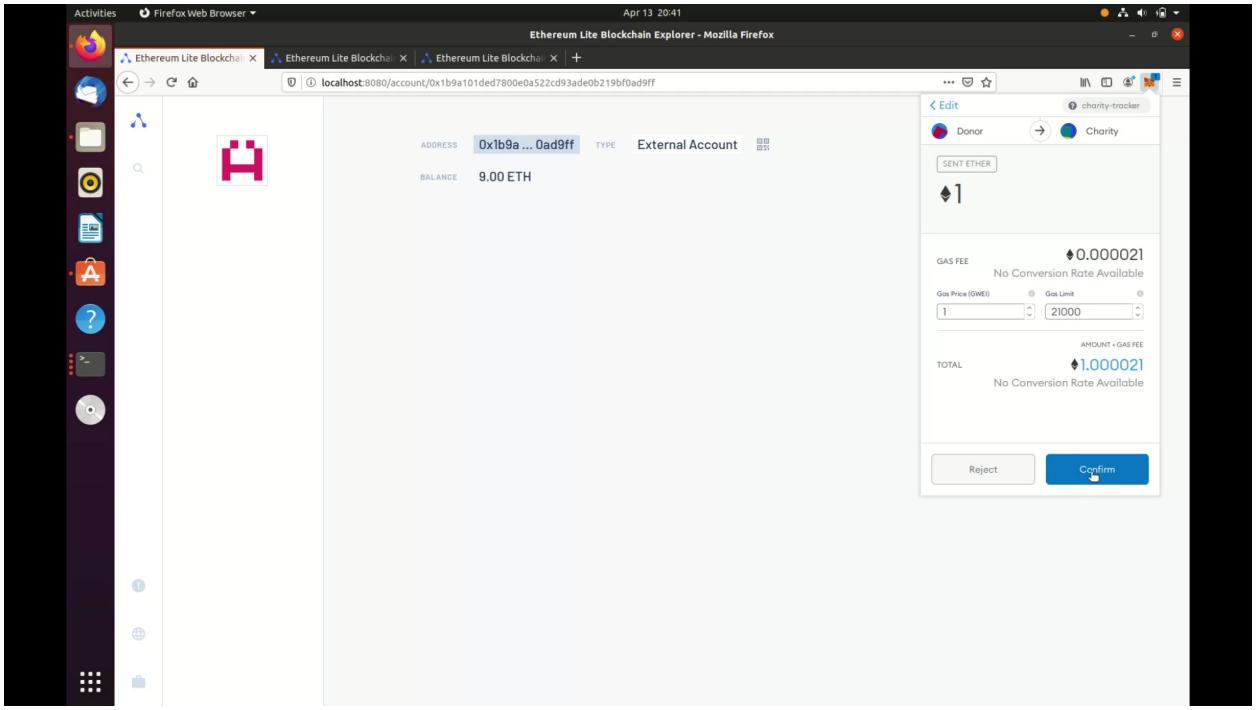


Fig 9. 1 ETH donated to Charity by Donor

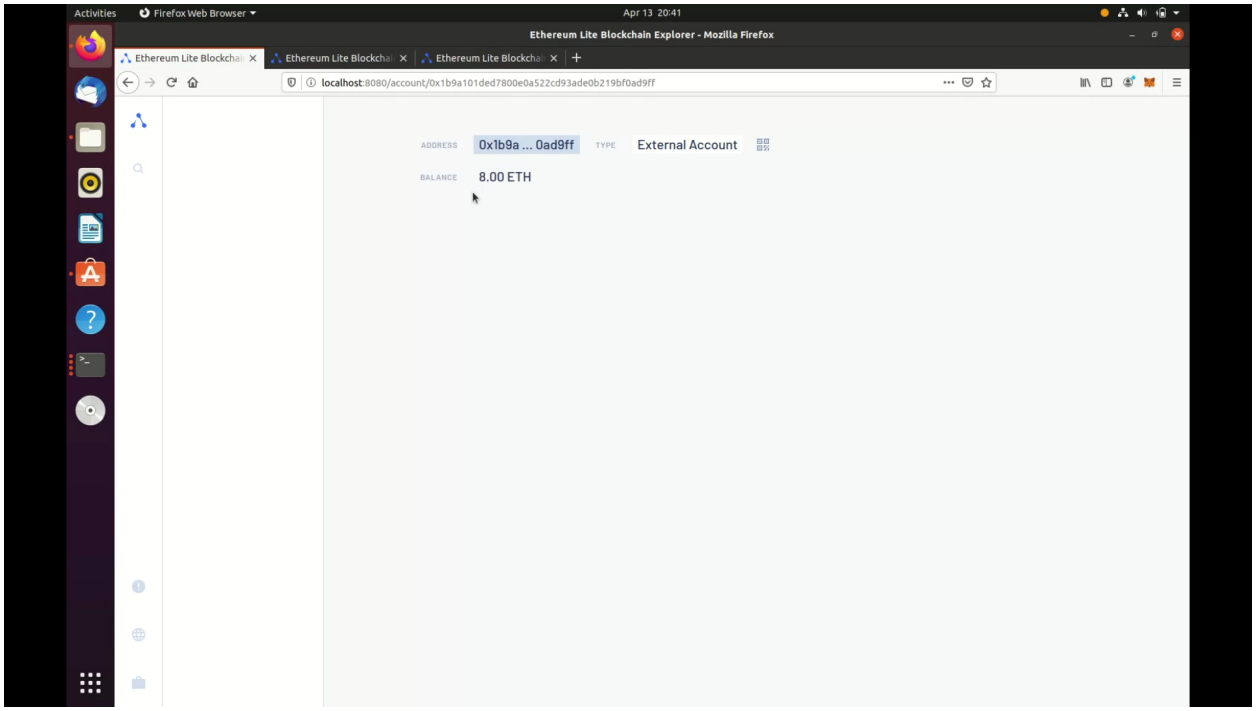


Fig 10.a

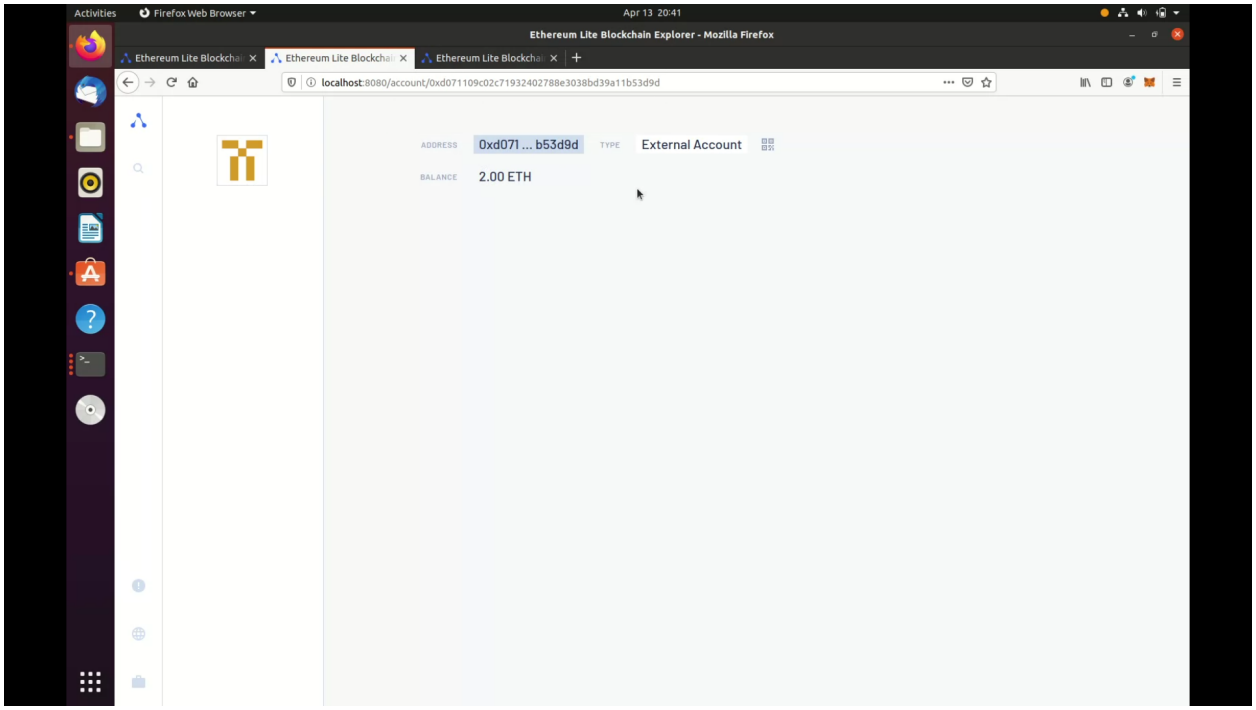


Fig 10.b

Fig. 10.a and 10.b show confirmation of donation. Donor has 8 ETH and Charity has 2 ETH after the transaction

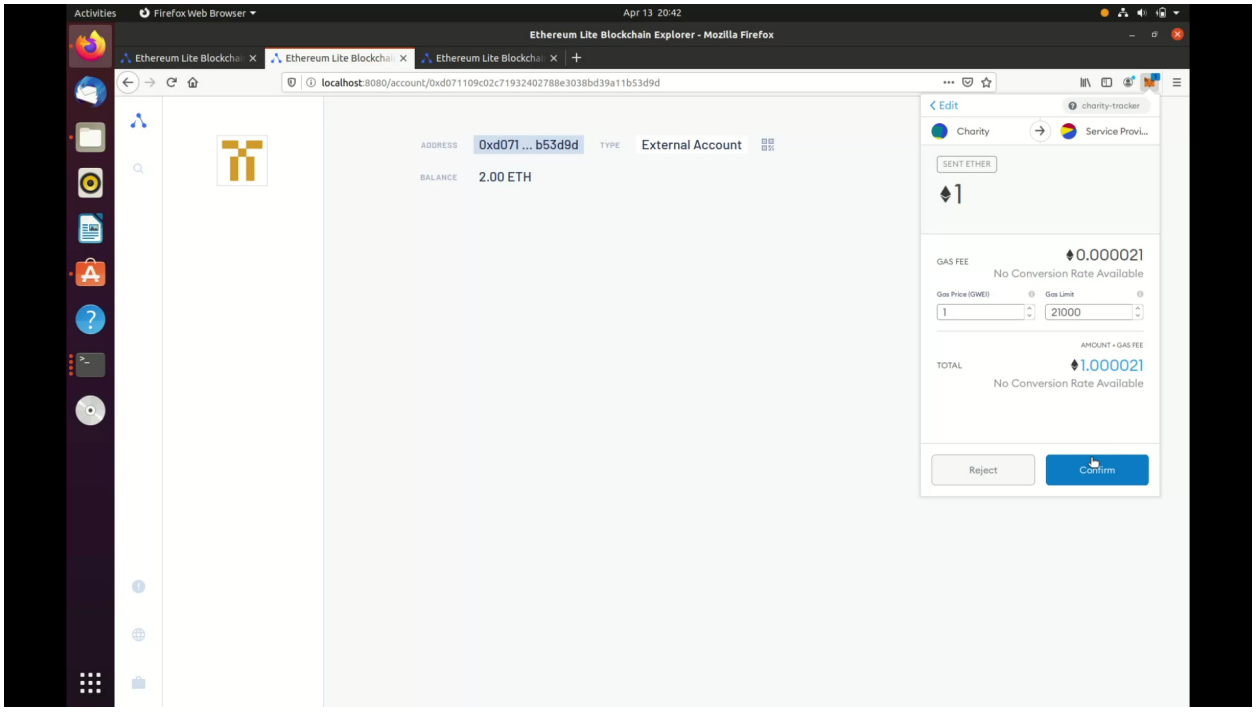


Fig 11. 1 ETH paid by Charity to Service Provider

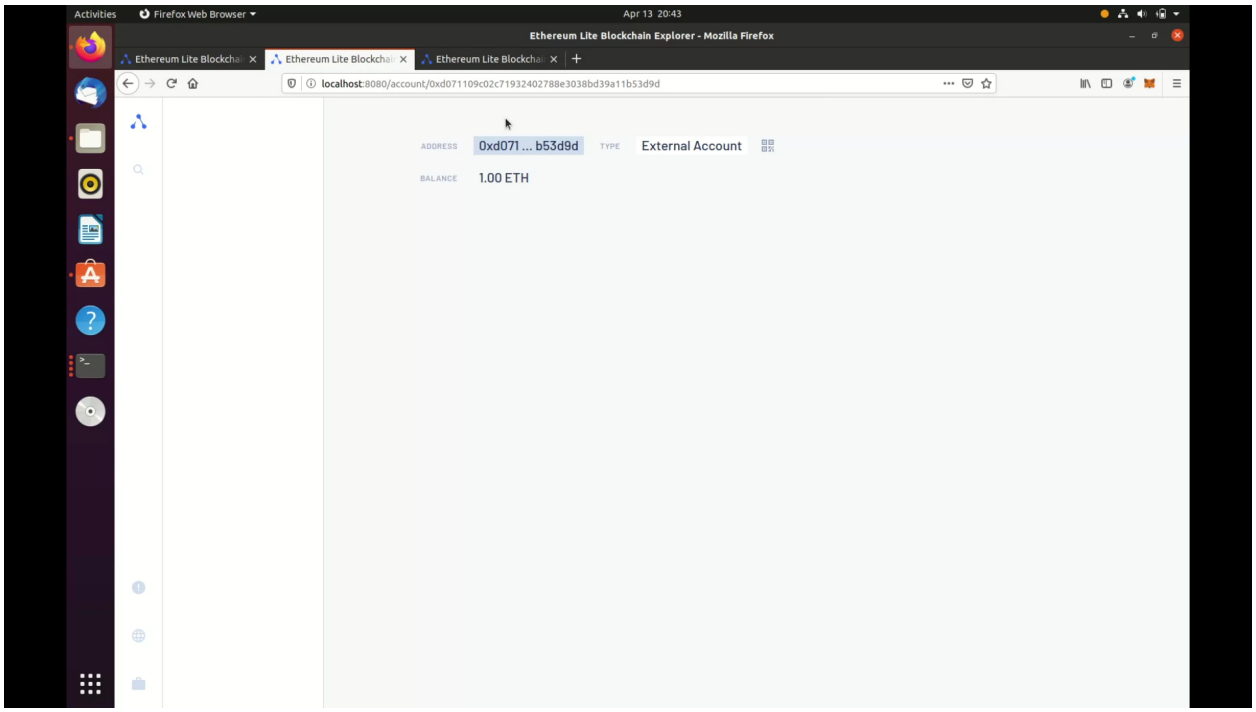


Fig 12.a

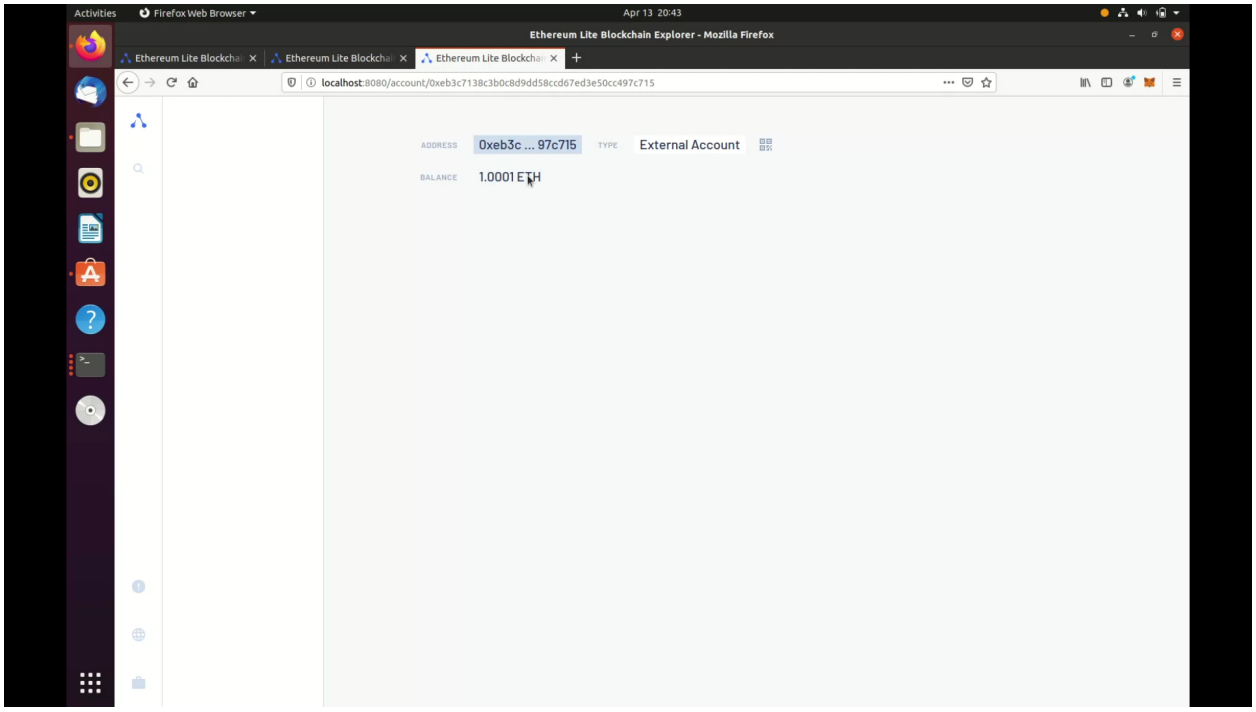


Fig 12.b

Fig. 12.a and 12.b show confirmation of payment. Charity has 1 ETH and Service Provider has 1 ETH after the transaction

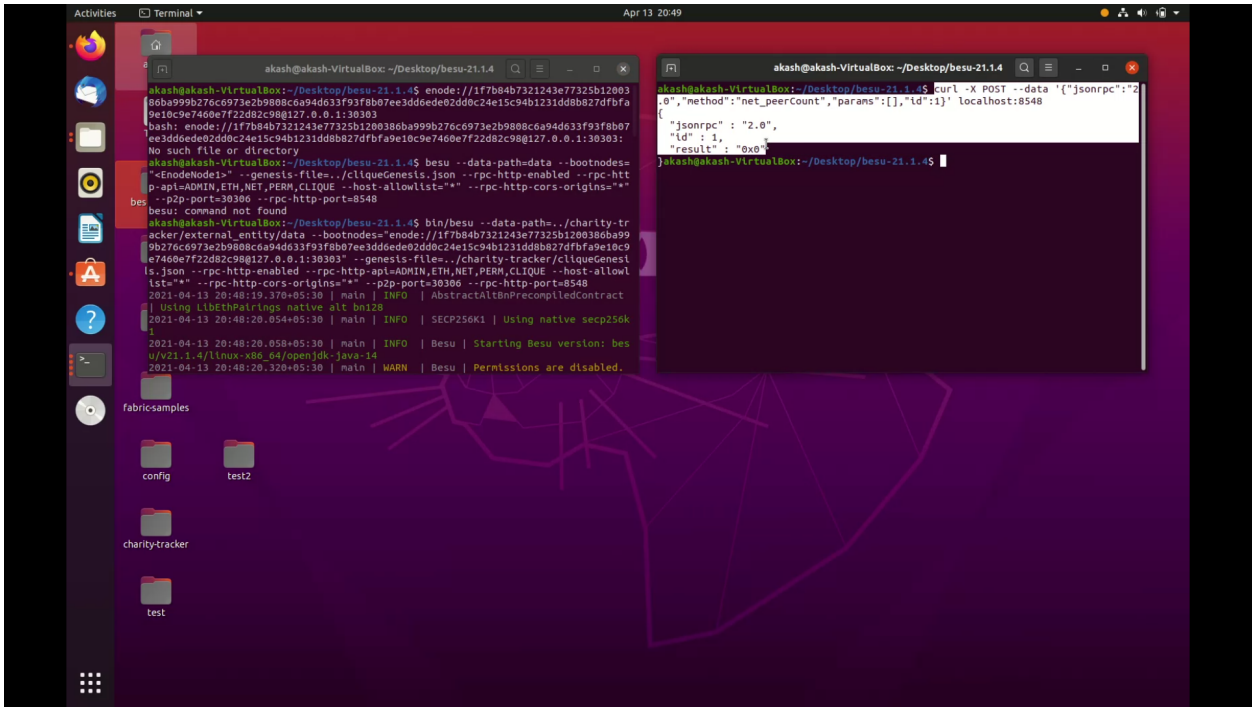


Fig. 13. In the lhs terminal an ‘external_entity’ node is started which isn’t in the allowed-list. The result on the rhs terminal confirms that this node has no peers despite specifying ‘charity’ as bootnode.

Advantages of the project over the traditional solutions commercially available

This project integrates the permissioning and private network properties of hyperledger and command line interface & JSON-RPC API for running, maintaining, debugging, and monitoring nodes in an Ethereum network. The traditional solutions available that use ethereum fail to provide permissioning to the private network whereas the solutions implemented on hyperledger fabric struggle to maintain the chaincode and standard cryptocurrency. So, this project has taken benefits of both platforms wherein we make a private permissioned network on hyperledger besu and send transactions through metamask in ethereum. Also Alethio ethereum lite explorer helps the nodes to monitor balance of various addresses. All donors have access to the address of charity and service organisations and can review their balances and transactions using this explorer which accounts for complete transparency.

Scope for improvement

Due to time constraint, we could not implement all our objectives. This project can include multi-signature concepts in smart contracts where a transaction signed by charity will only be approved when the service provider also signs on it. This helps us to ensure that money was actually received by the service provider (in some cases directly to the beneficiaries). Physical evidence can also be incorporated by the charity organisation which involves timely updates to its donors through photographs, videos, etc. Also we realised that in today's scenario not everybody is familiar with cryptocurrency. All our transactions are based purely on ethereum and there would need of exchange organisations to convert it into fiat currency on various occasions.

Experience with the project

This project was a good learning experience for us where we could actually implement the concepts learned in class. Our problem statement required implementation of a private permissioned network for charity organisation. So, we decided to go with hyperledger fabric and composer initially. But, we learnt that hyperledger composer had been deprecated. Then we tried our hands with fabric but we realised it would be difficult to write the entire chaincode and keep a standard for cryptocurrency. Then, we came across hyperledger besu which combined the permissioning and private

network of hyperledger with the ease of maintaining information about transactions on Metamask. This solved a lot of our problems. The formal documentation on hyperledger.org was of great help.

Despite these challenges, our project is successful in running a private permissioned network, making transactions on metamask and viewing their state on block explorer.