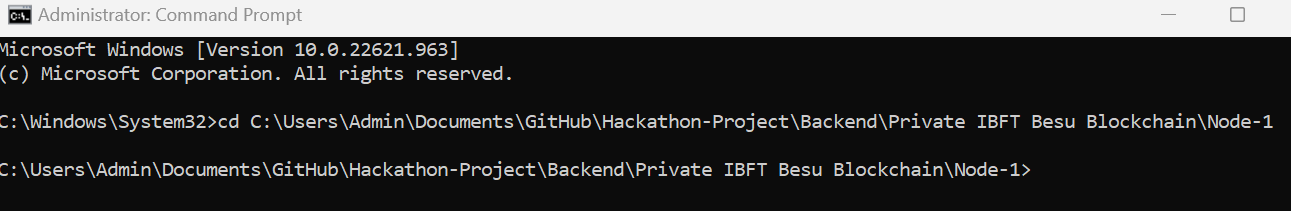
Hackathon Project: Digital ID Documentation

1. Run the Private Blockchain

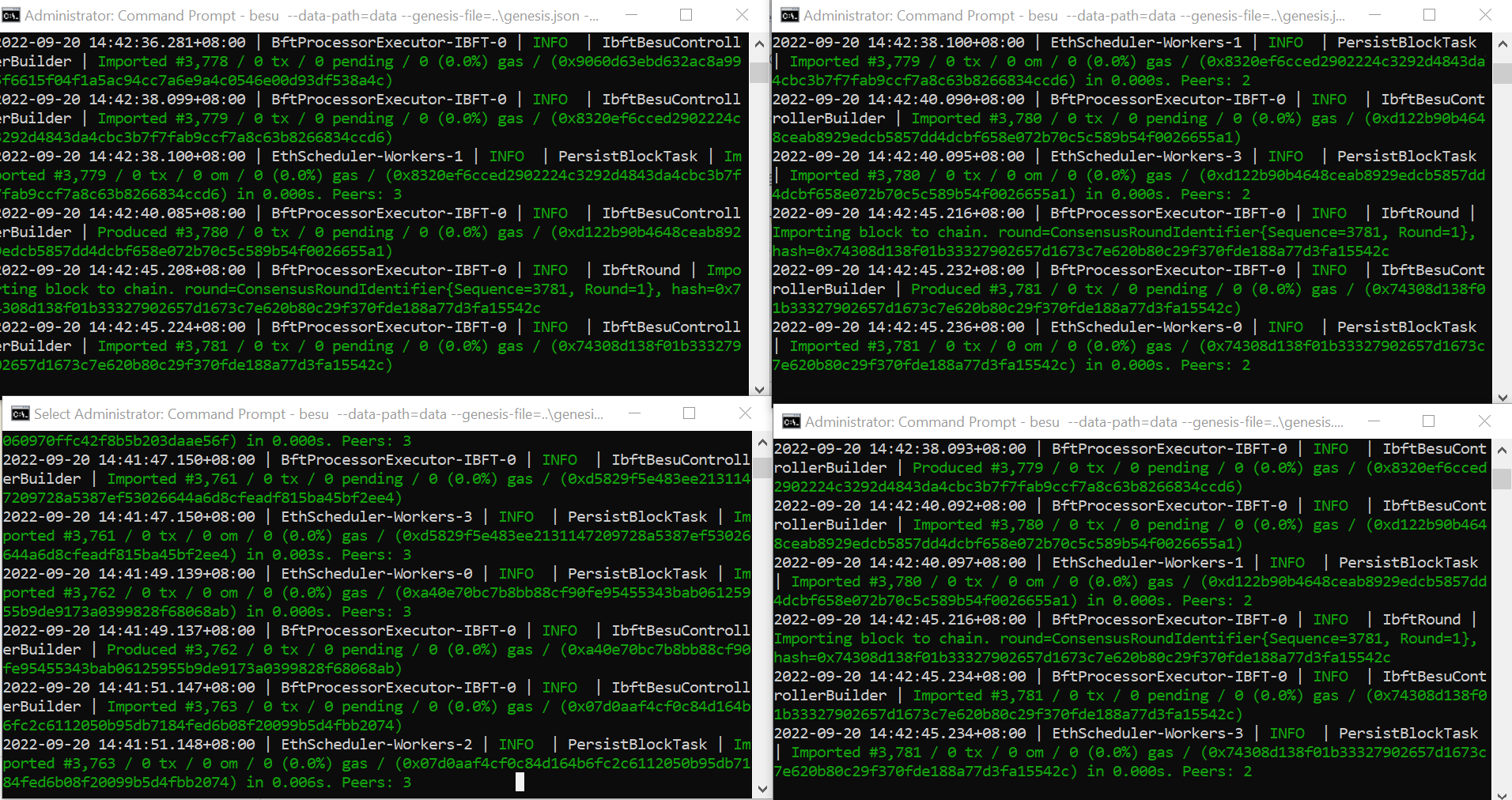
* Grab the file paths for the 1st to 4th node respectively under the Private IBFT Besu Blockchain Folder



* Open up 4 command prompts with administrator and cd into all 4 of the file paths (from Node 1 to 4)

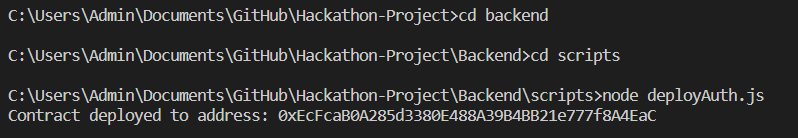


* Copy and paste the following Code to run the Private Blockchain in each of their respective directories
  + Node 1: besu --data-path=data --genesis-file=..\genesis.json --rpc-http-host=0.0.0.0 --rpc-http-enabled --rpc-http-api=ETH,NET,IBFT,EEA,PRIV --host-allowlist="\*" --rpc-http-cors-origins="all" --privacy-enabled --privacy-url=http://127.0.0.1:9102 --privacy-public-key-file=Tessera\nodeKey.pub --min-gas-price=0 --rpc-ws-enabled=true --rpc-ws-host=0.0.0.0 --rpc-ws-port=9545
  + Node 2: besu --data-path=data --genesis-file=..\genesis.json --bootnodes=enode://a32b7b959c0acc4fda78814f1261ab964b535a7e63f25f9fd149b391a016297c7f8b200b9e8a90790904ac0e68ff9c31c7fae63dca866a2e4080a314c7bebe86@127.0.0.1:30303 --p2p-port=30304 --rpc-http-host=0.0.0.0 --rpc-http-enabled --rpc-http-api=ETH,NET,IBFT,EEA,PRIV --host-allowlist="\*" --rpc-http-cors-origins="all" --rpc-http-port=8546 --privacy-enabled --privacy-url=http://127.0.0.1:9202 --privacy-public-key-file=Tessera\nodeKey.pub --min-gas-price=0 --rpc-ws-enabled=true --rpc-ws-host=0.0.0.0 --rpc-ws-port=9546
  + Node 3: besu --data-path=data --genesis-file=..\genesis.json --bootnodes=enode://a32b7b959c0acc4fda78814f1261ab964b535a7e63f25f9fd149b391a016297c7f8b200b9e8a90790904ac0e68ff9c31c7fae63dca866a2e4080a314c7bebe86@127.0.0.1:30303 --p2p-port=30305 --rpc-http-host=0.0.0.0 --rpc-http-enabled --rpc-http-api=ETH,NET,IBFT,EEA,PRIV --host-allowlist="\*" --rpc-http-cors-origins="all" --rpc-http-port=8547 --privacy-enabled --privacy-url=http://127.0.0.1:9302 --privacy-public-key-file=Tessera\nodeKey.pub --min-gas-price=0 --rpc-ws-enabled=true --rpc-ws-host=0.0.0.0 --rpc-ws-port=9547
  + Node 4: besu --data-path=data --genesis-file=..\genesis.json --bootnodes=enode://a32b7b959c0acc4fda78814f1261ab964b535a7e63f25f9fd149b391a016297c7f8b200b9e8a90790904ac0e68ff9c31c7fae63dca866a2e4080a314c7bebe86@127.0.0.1:30303 --p2p-port=30306 --rpc-http-host=0.0.0.0 --rpc-http-enabled --rpc-http-api=ETH,NET,IBFT,EEA,PRIV --host-allowlist="\*" --rpc-http-cors-origins="all" --rpc-http-port=8548 --privacy-enabled --privacy-url=http://127.0.0.1:9402 --privacy-public-key-file=Tessera\nodeKey.pub --min-gas-price=0 --rpc-ws-enabled=true --rpc-ws-host=0.0.0.0 --rpc-ws-port=9548
* After a few seconds, the blocks in the private blockchain should start validating which ensures that all 4 nodes are functioning



1. Deploy the Auth.sol mapping contract

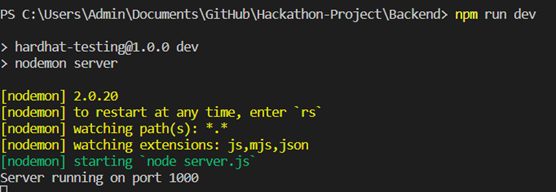
* Since the Private Blockchain will be running on a new device, it will not recognise the already deployed Auth.sol Smart Contract since it does not exist. Hence, we would need to deploy it once
* cd into the Backend and then into scripts and run the deployAuth.js script with “node deployAuth.js” and copy the smart contract address when deployed



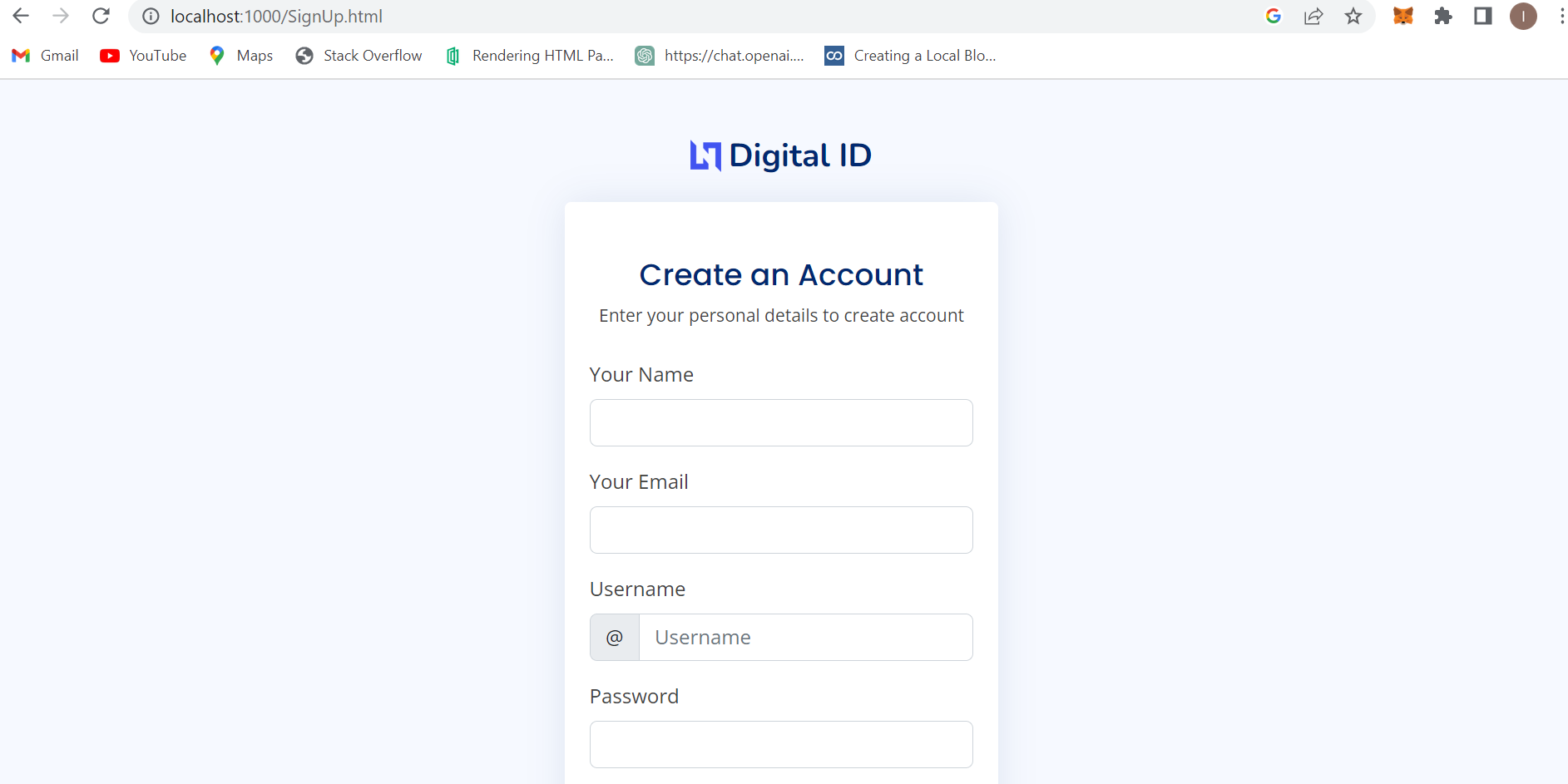
* Paste the deployed smart contract address into server.js and replace the address with the address present in “const auth = new ethers.Contract("0x0d8cc4b8d15D4c3eF1d70af0071376fb26B5669b", contract.abi, signer);” at line 9



1. Run the server
   * To run the server, cd into the Backend and type “npm run dev” in the terminal, and head to localhost:1000 on your device.



* + Congrats! Now you should be redirected to the Sign-Up Page where all the data submitted will be rendered into the Auth.sol mapping smart contract.



* + After Signing up with your name, email, username and password. You could also sign out and then log in through the website again as all the data will be stored into the Ethereum Smart Contract on the Private Blockchain!

