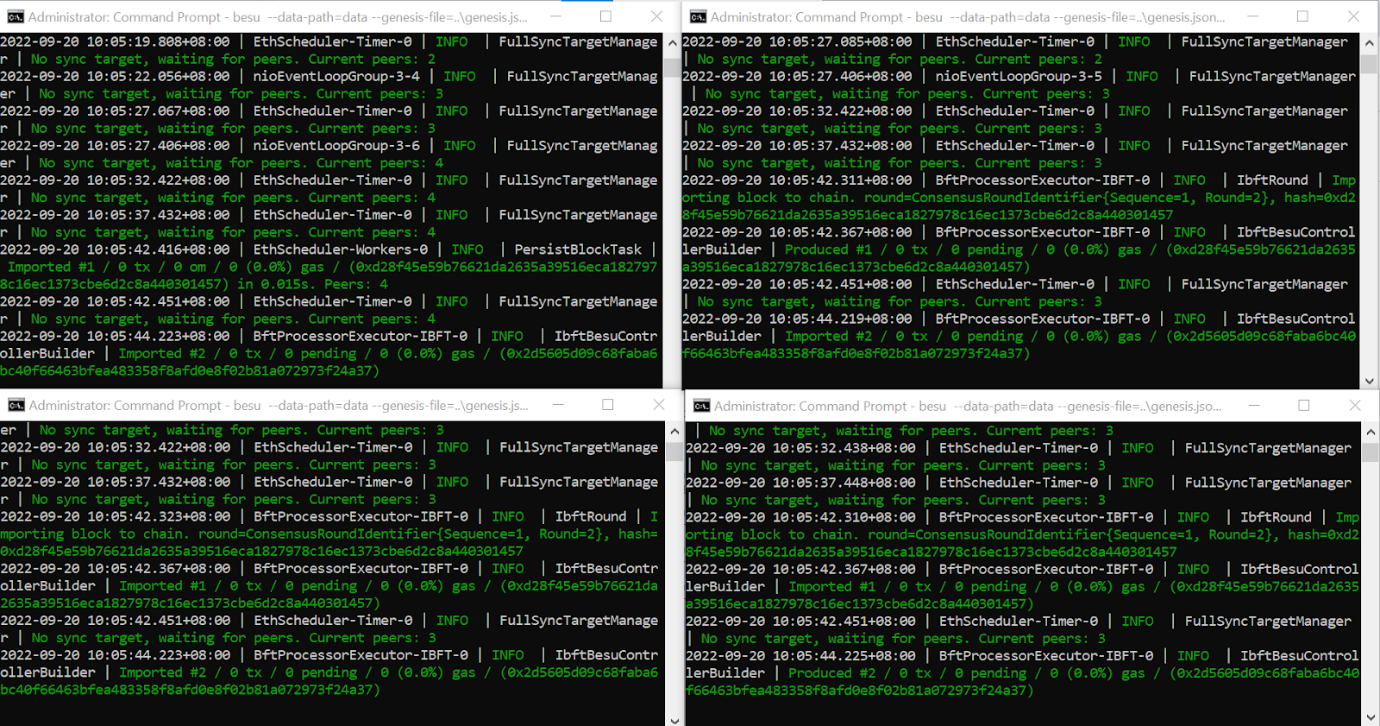
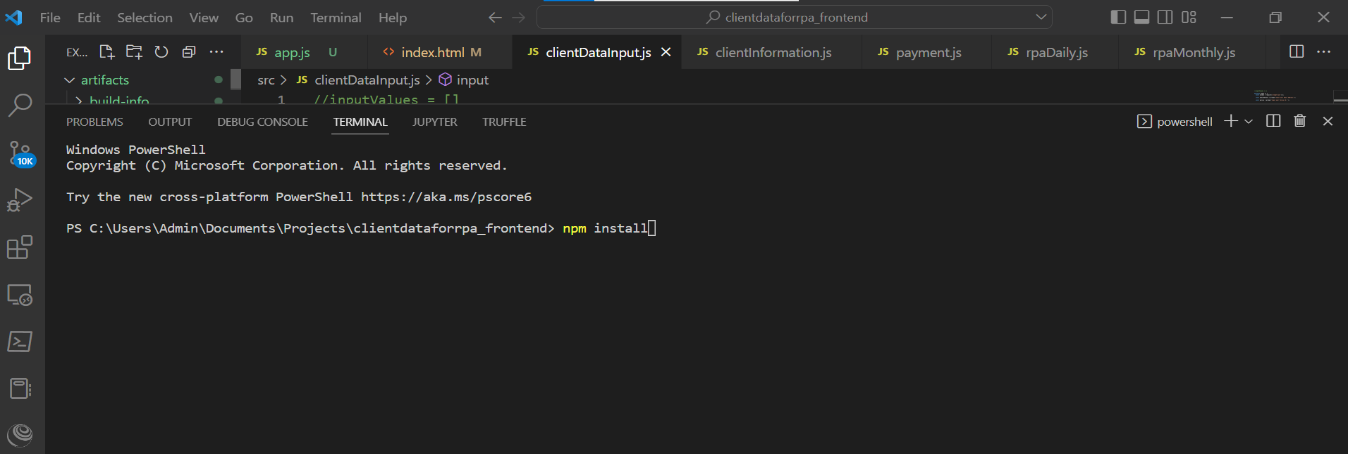
Presentation Demo Setup

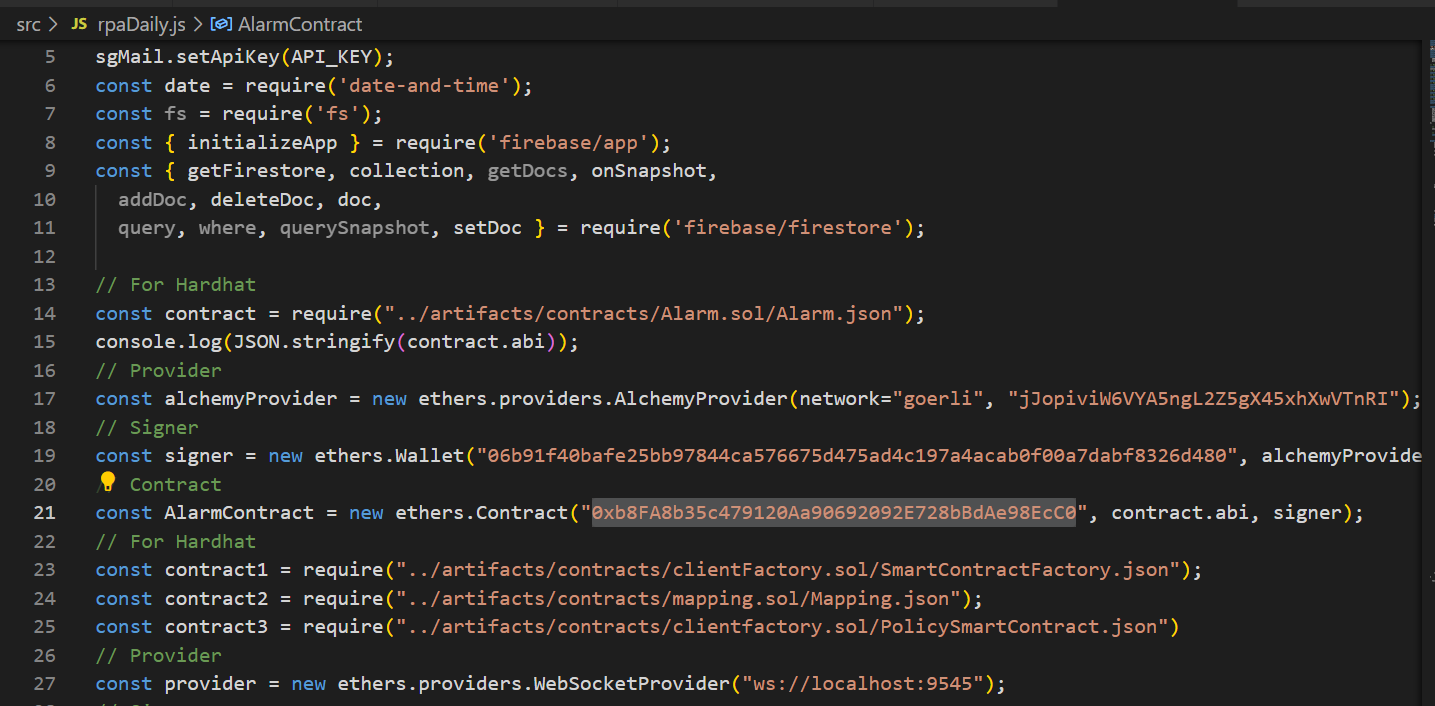
* Run the Private Blockchain with the following commands
  + Node 1 - besu --data-path=data --genesis-file=..\genesis.json --rpc-http-host=0.0.0.0 --rpc-http-enabled --rpc-http-api=ETH,NET,IBFT,EEA,PRIV --host-allowlist="\*" --rpc-http-cors-origins="all" --privacy-enabled --privacy-url=http://127.0.0.1:9102 --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

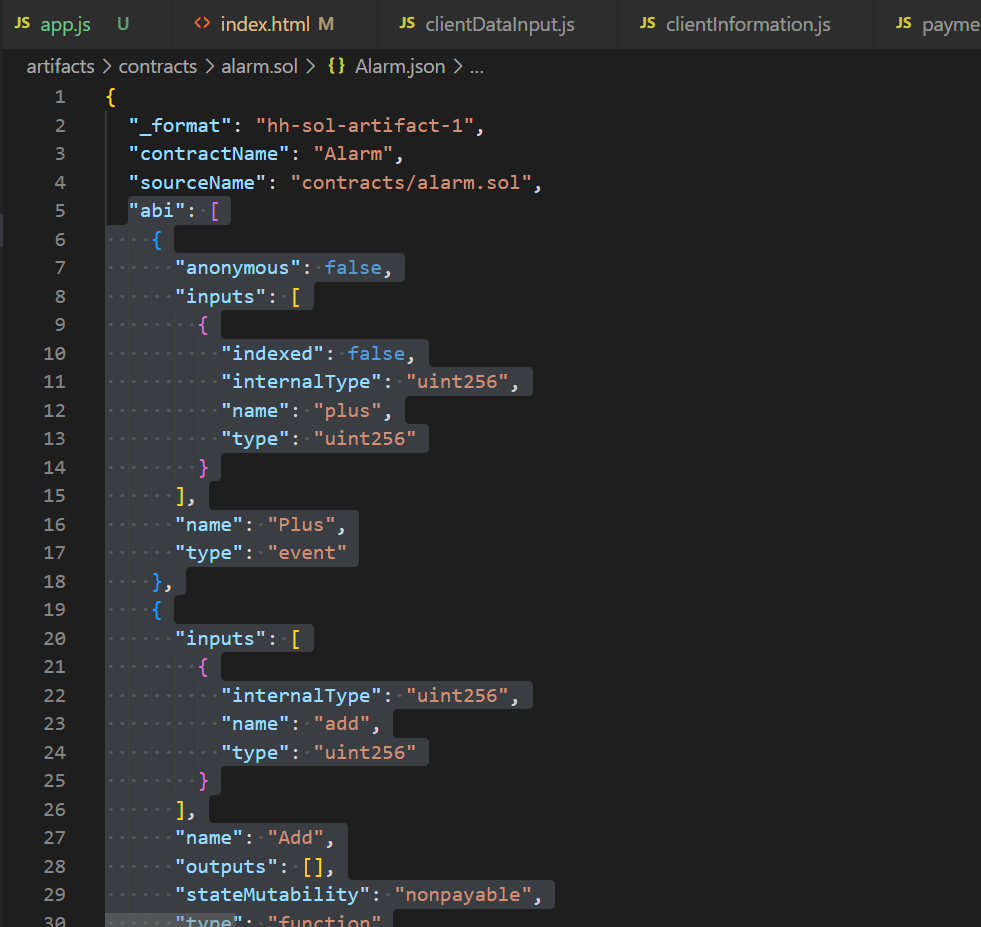


* Setup the Gelato Network
  + Open up the clientdataforrpa\_frontend folder on VsCode and run “npm install”

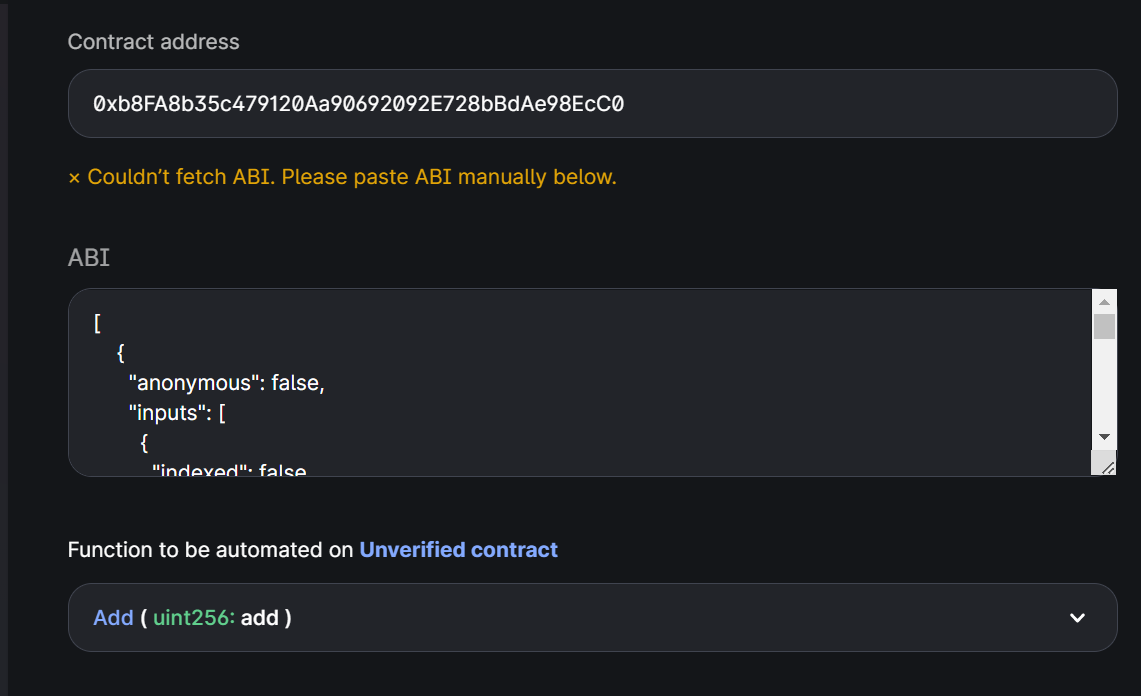


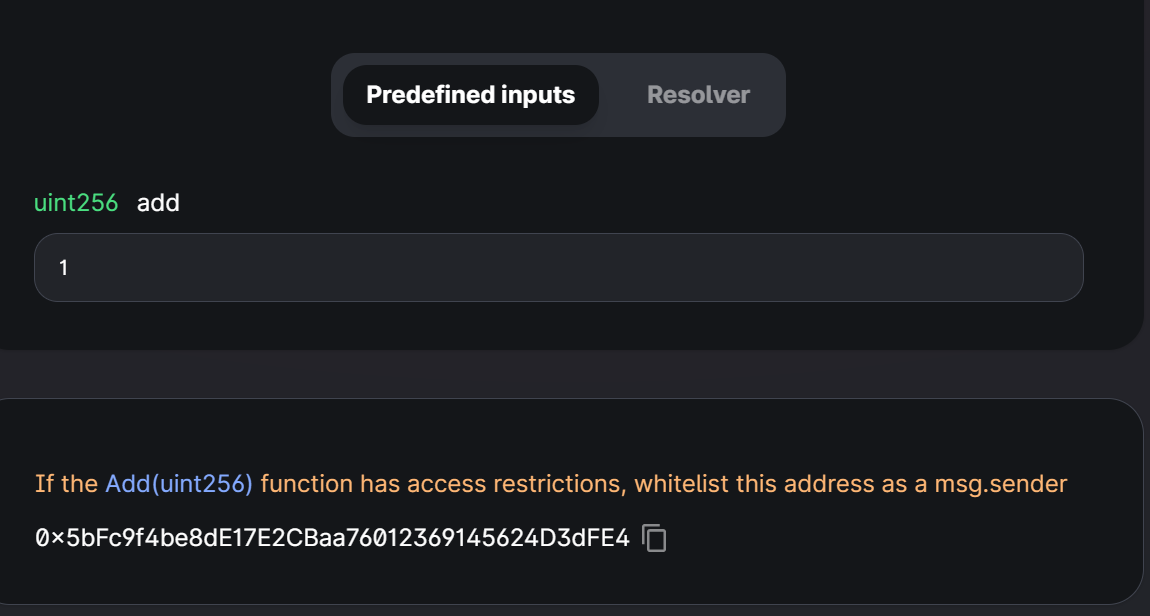
* + Head to rpaDaily.js and copy the contract address and abi of alarm.sol

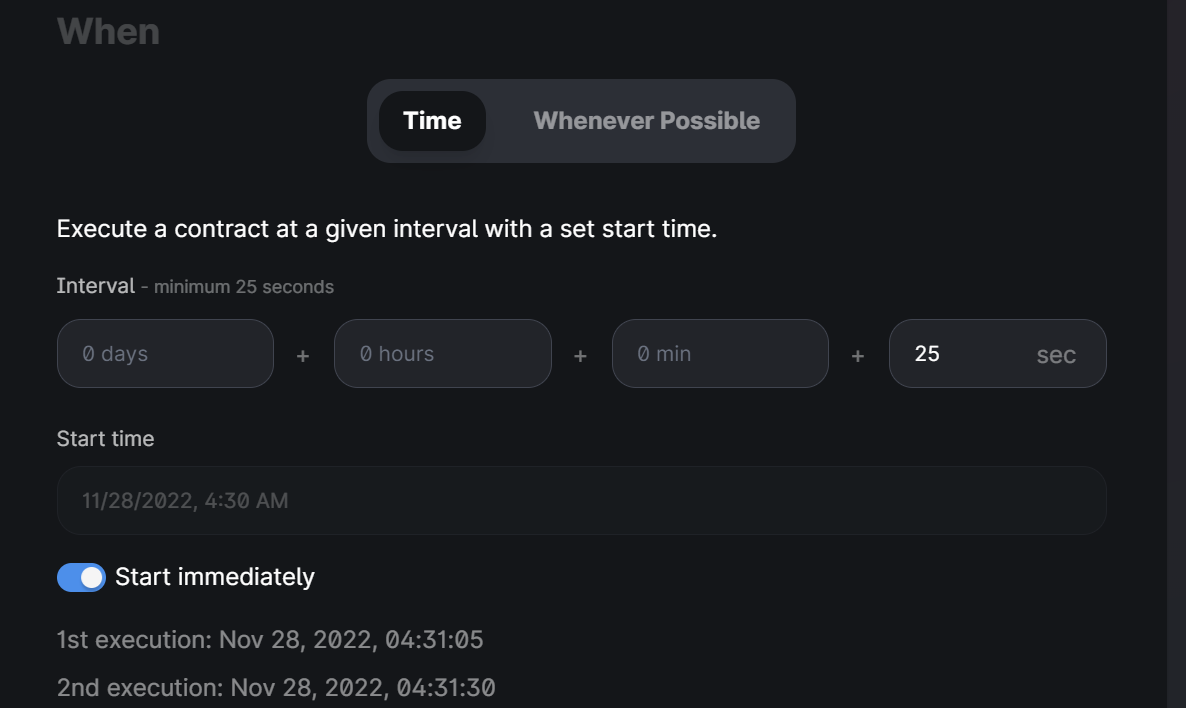


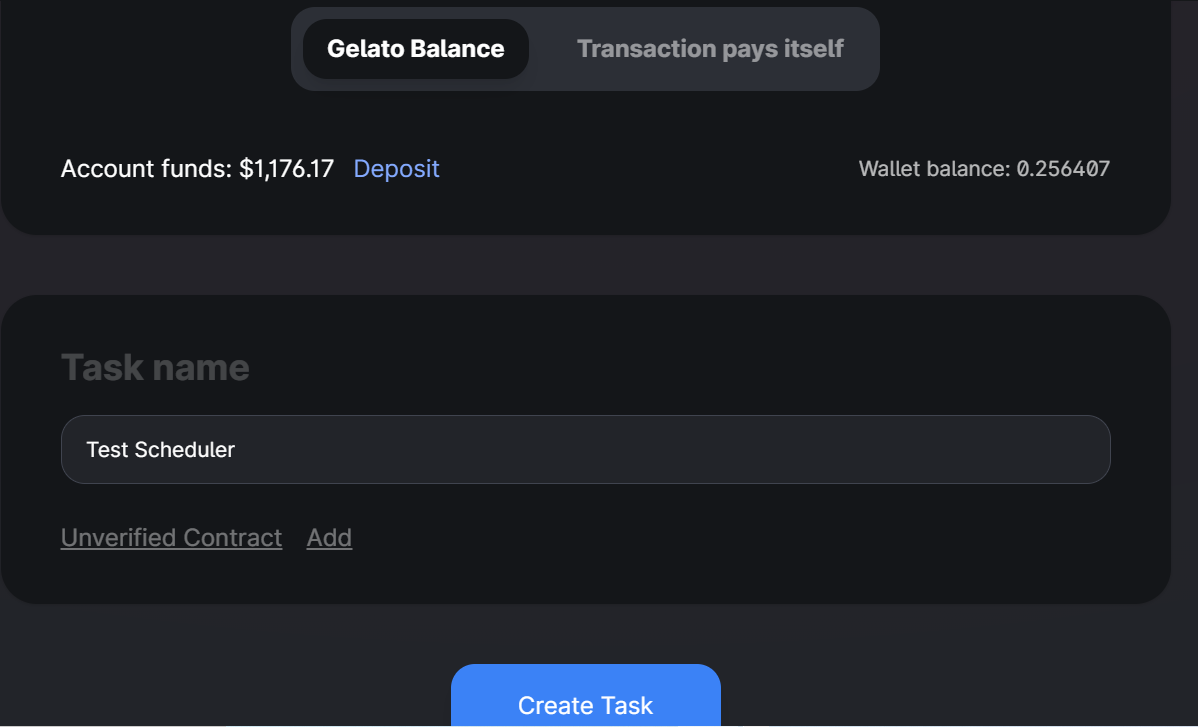


* + Head to the Gelato Automation Website (<https://app.gelato.network/>) and create a new Task with the contract address and the following details

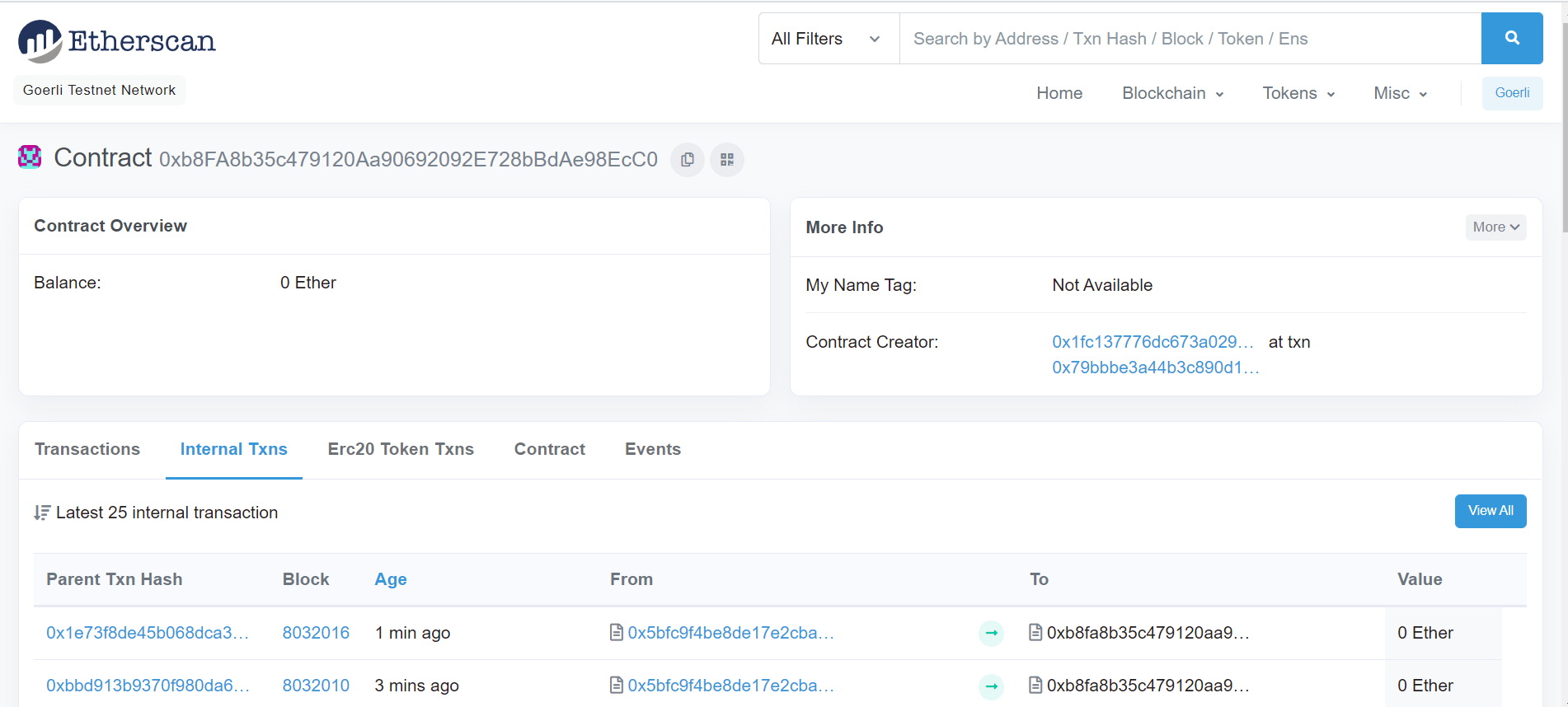




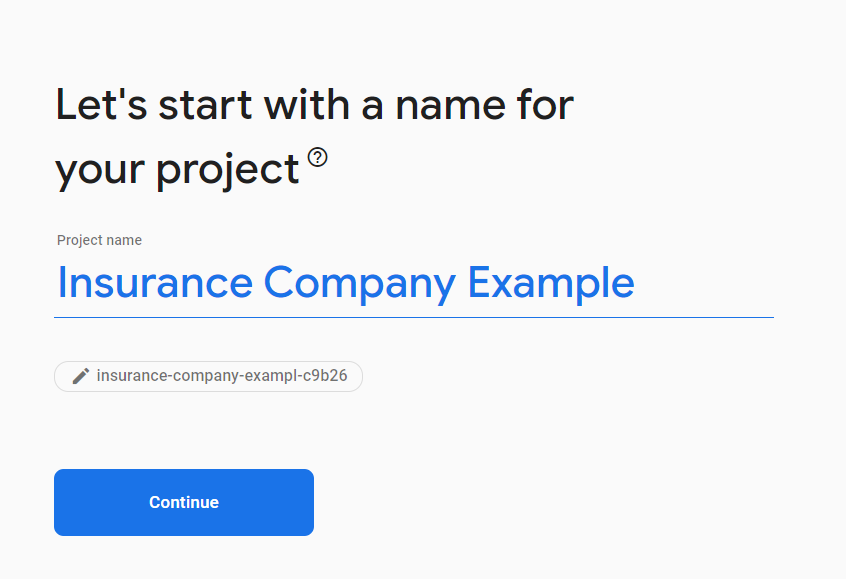


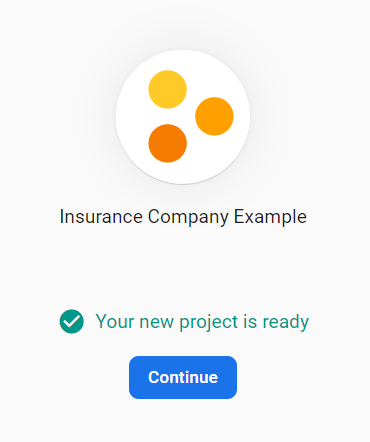


* + Once its executed click on the contract Address and check for the internal transactions under Goreli Etherscan

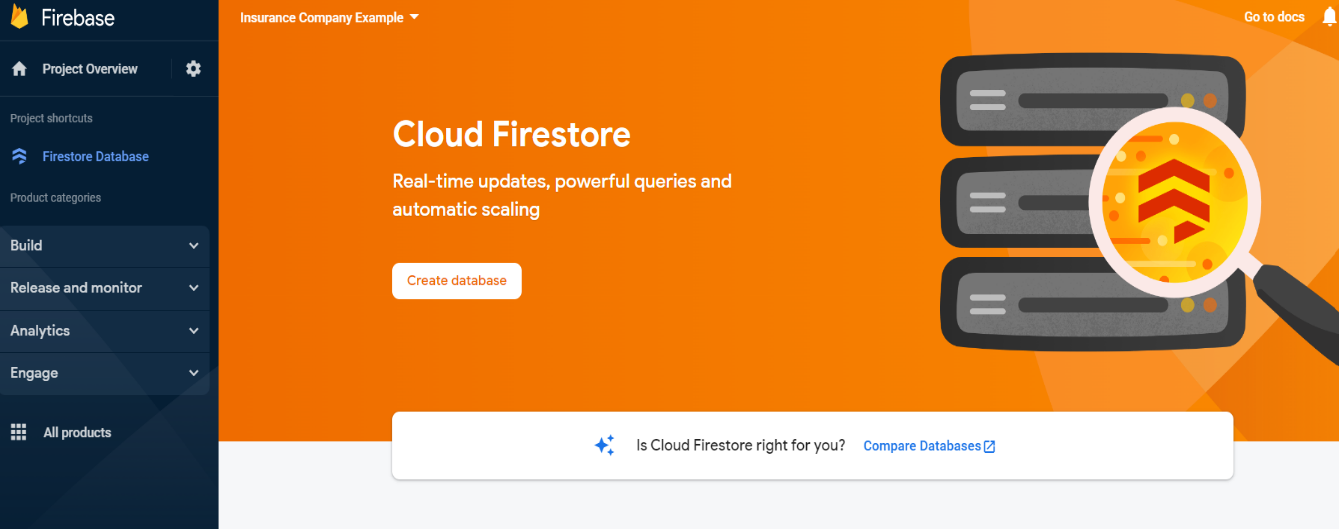


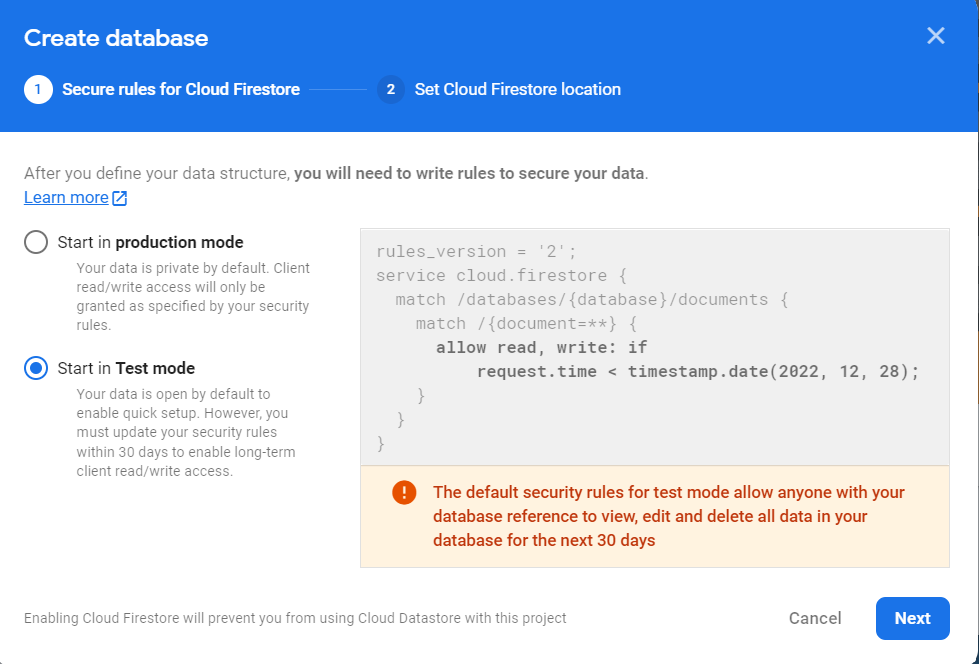
* Setup the Firebase Database
  + Click on Go to Console and Create a New Project with the name Insurance Company (Rest of the Options are left on default)

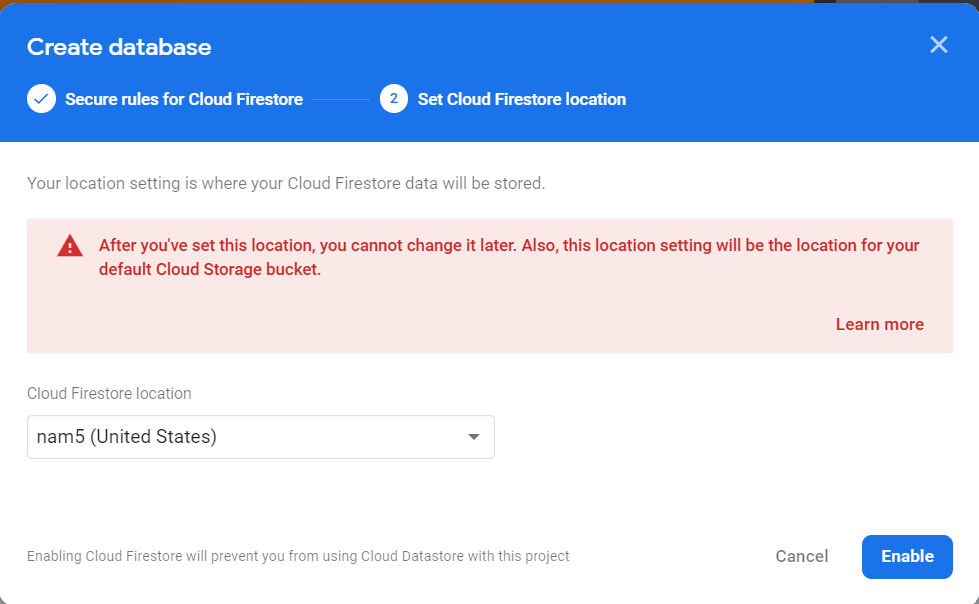


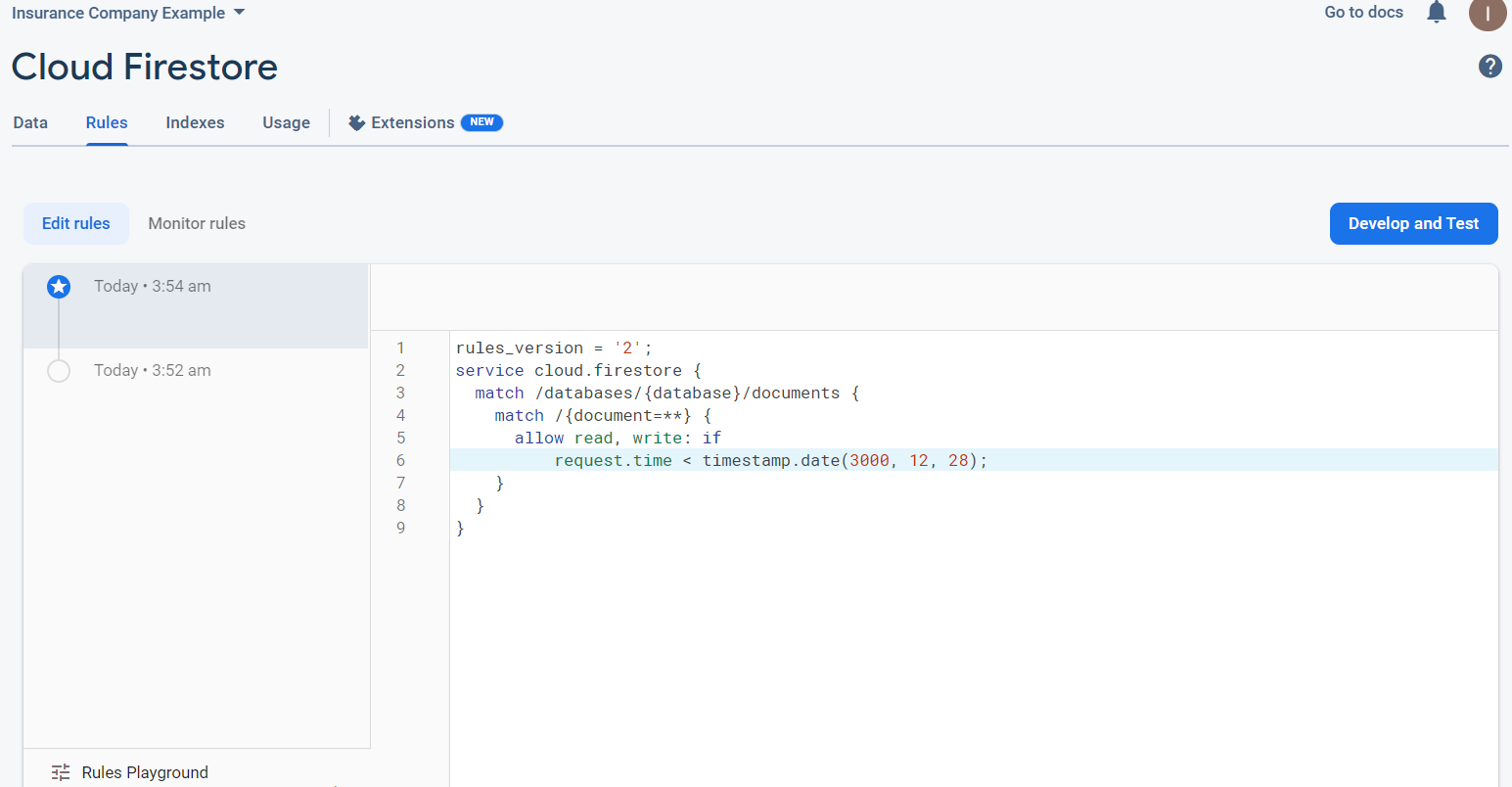


* + Click on Firestore Database and create a new Database with the following options and rules

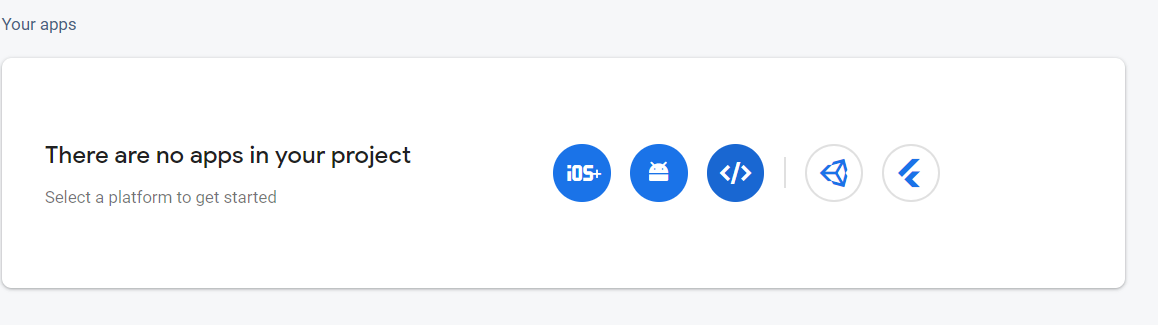


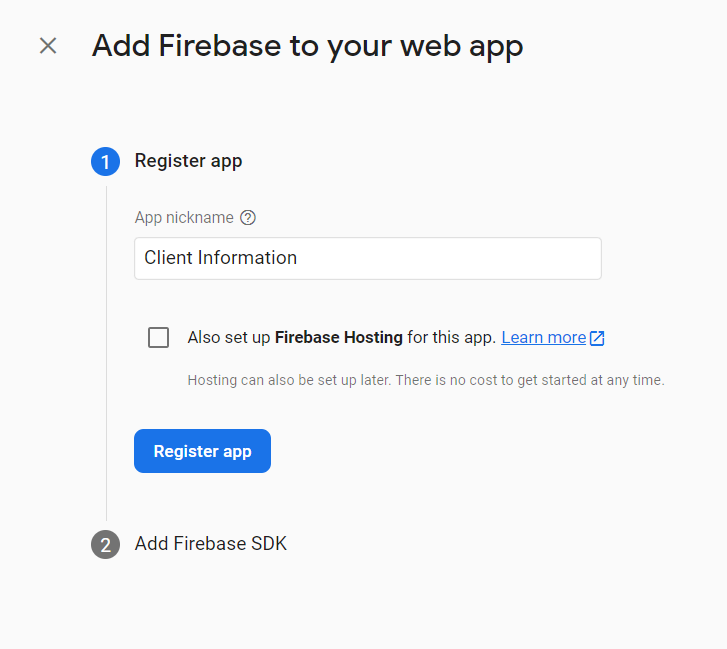


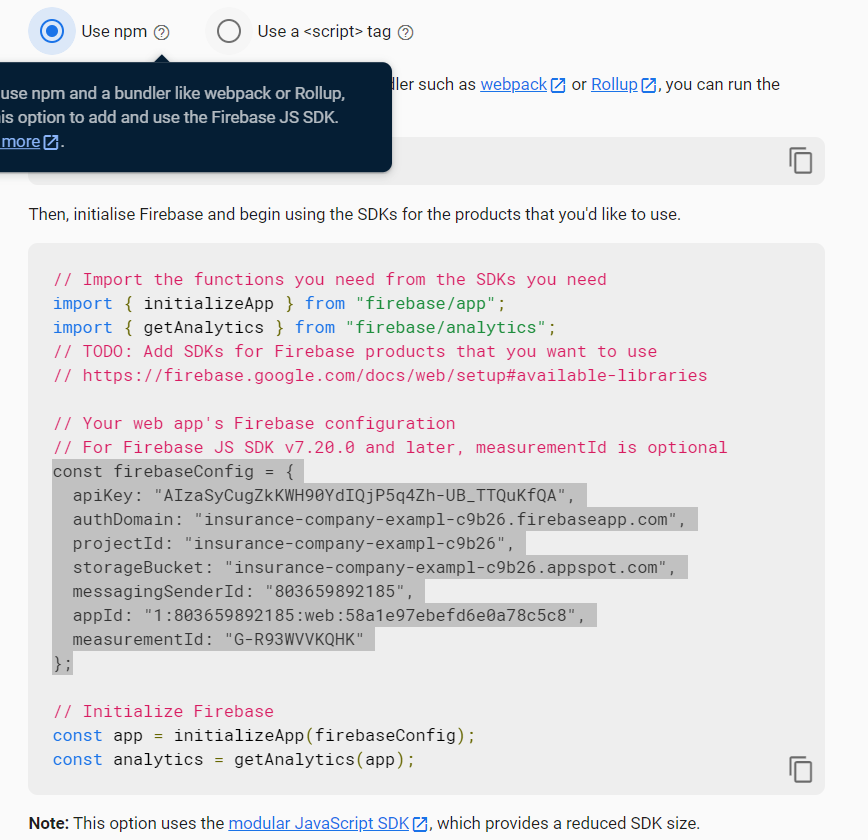


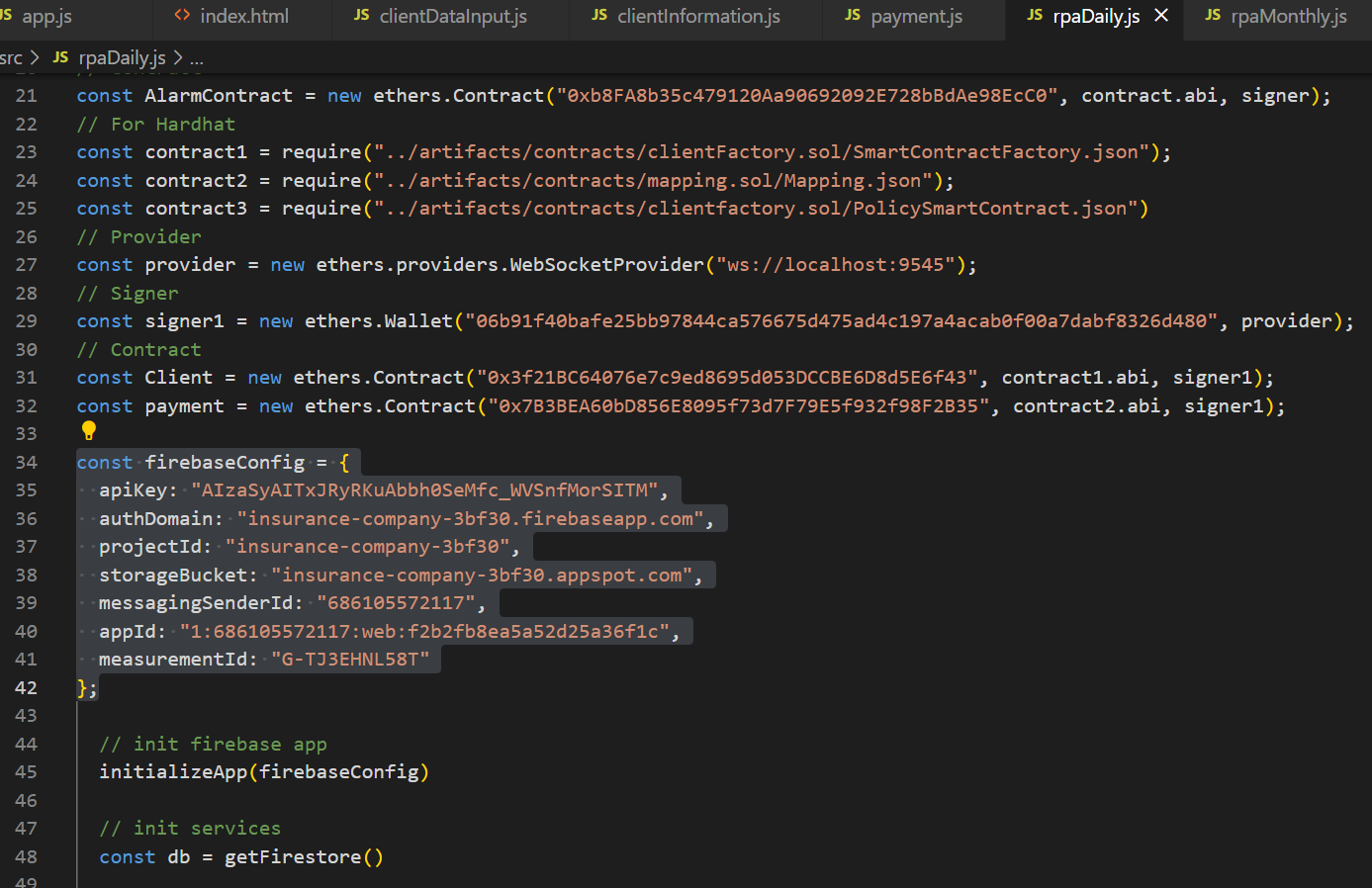


* + Click on Project Settings to Copy the Firebase Config and paste it into rpaDaily.js in the clientforrpa\_frontend folder in VsCode

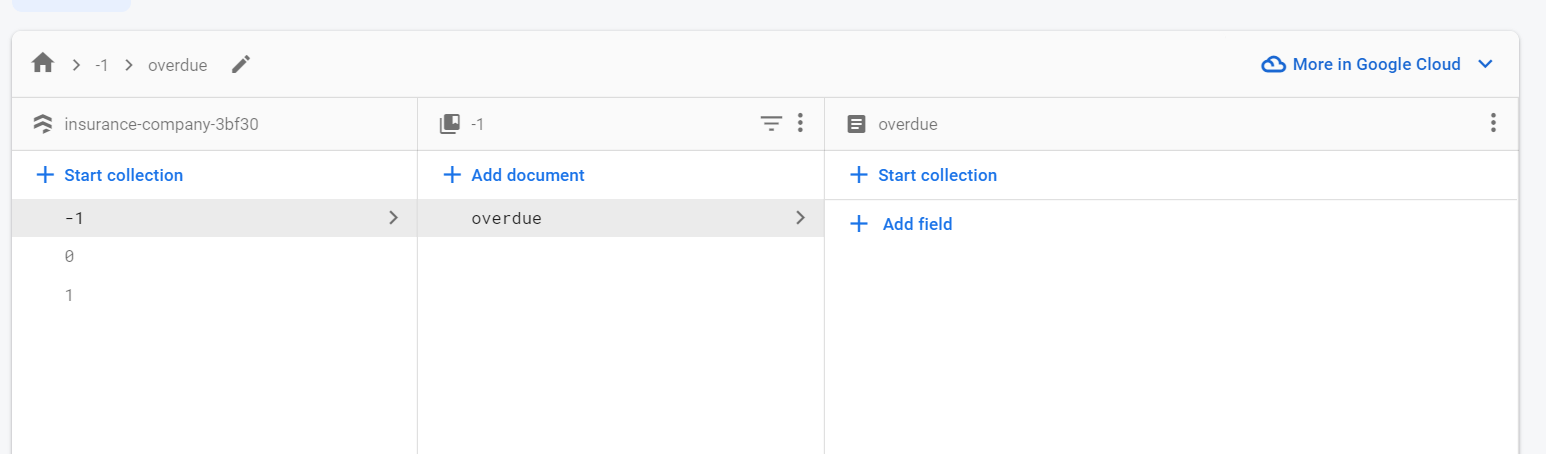




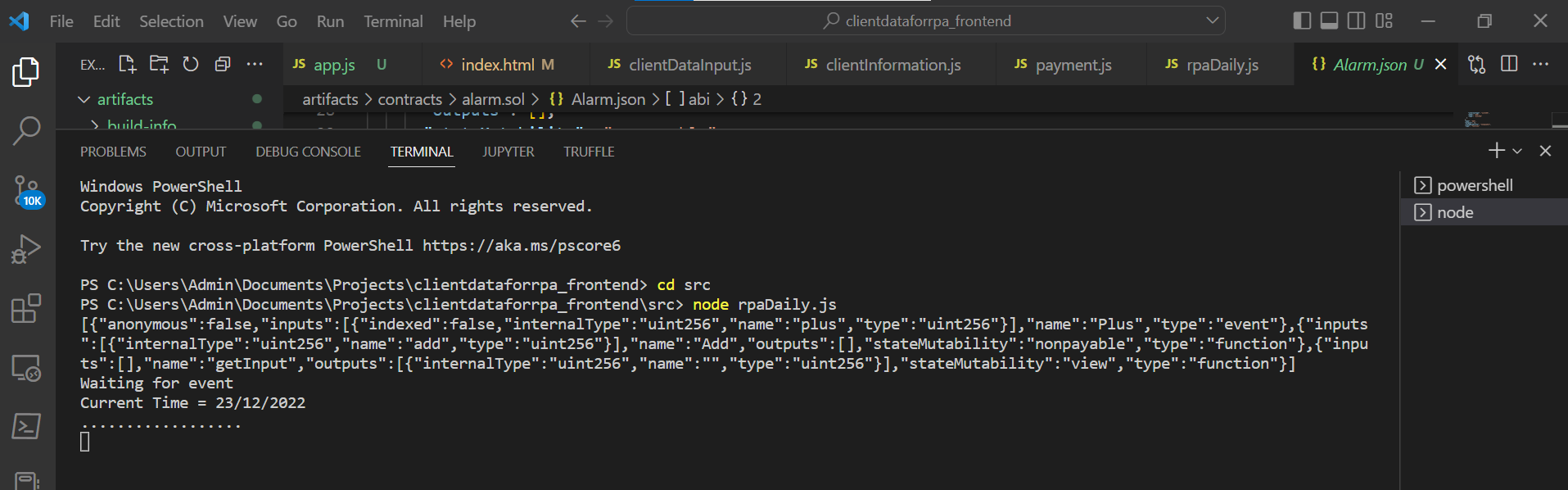




* + Create the Database Structure in the form:
    - -1 (Collection) => overdue (document id)
    - 0 (Collection) => unpaid (document id)
    - 1 (Collection) => paid (document id)



* Run the Following Backend Files in the following order
  + Run rpaDaily.js with the following commands



* + Run clientinformation.js and payment.js with the same format

