Review Open Call F4Fp-SME (Stage 1) SO-SHARED experiment

Mandana Falahi

BEIA CONSULT INTERNATIONAL

Remote Review

May 2022

Concept

SO-SHARED (BLOCKCHAIN-ENABLED SHARING ECONOMY PLATFORM FOR COMMUNITIES)

- Many organizations are pressured nowadays to come across with innovative, technologically advanced products and services with low budgets
- Thus, resources must be completely used and focused on the most noteworthy priorities at any given time.
- Unluckily, inefficient or below the standard resource management will absolutely lead to unfavorable results, such as poor productivity, delays, decreased quality, increased costs, etc.
- Resources like services, tools and appliances (STA) are not used to their maximum capacity and detract economic productivity, leading to slowly amortizing investments, high levels of waste and social disintegration of communities.

Objectives

- SO-SHARED presents a social sharing platform through which users share STAs to build communities and create value.
- We propose a solution for the better utilization of physical resources within smart communities.
- The challenge is to unlock this potentially vast marketplace and create significant improvements in utilization of resources, community cohesion, costs reduction and more.
- The platform will match supply and demand based on geolocation and customer preferences with proven technology. Sharing conditions are recorded on a blockchain. The implemented services will preserve the anonymity of transactions while on the other hand it will facilitate matchmaking among the resource providers and consumers.
- Tailored business models will be provided as Smart Contract templates to be used by the stakeholders of the SO-SHARED platform

Hyperledger Fabric - Experiment Description

Set up a Hyperledger Fabric network so that registered users in Fabric CA can submit and evaluate transactions in the Hyperledger.

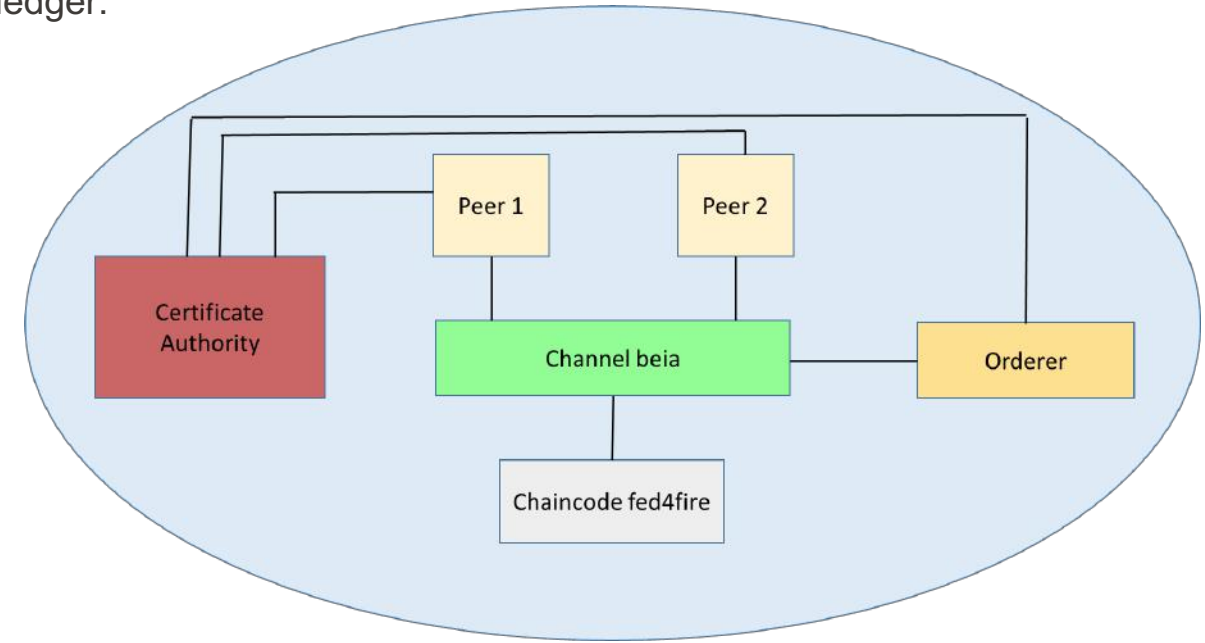
Channel name : beia

Chaincode : fed4fire

User : user01

Transaction :

(Project01 , Value01)



Install Hyperledger Fabric

PREQUISITION

Git

cURL

Node.js

Golang

Docker and Docker Compose

INSTALL FABRIC SAMPLES, BINARIES, AND DOCKER IMAGES

Fabric version : 2.2.2

Fabric CA version : 1.4.7

$curl -sSL https://bit.ly/2ysbOFE | bash -s -- 2.2.2 1.4.7

Starting Hyperledger Fabric

1. DEFINE A CHAINCODE

First Chiancode “fed4fire” is defined .Its includes three methods initledger, writeData and readData.When the network is first set up, the initledger is called up for initialization the ledger and ensuring that the chaincode is successfully installed.

```
'use strict';

const { Contract } = require('fabric-contract-api');

class Fed4Fire extends Contract {

async initLedger(ctx){
    await ctx.stub.putState("Beia", "Fed4Fire");
    return "success";
    }

    async writeData(ctx,key,value){
    await ctx.stub.putState(key,value);
    return value;

    }

    async readData(ctx,key){
    var response = await ctx.stub.getState(key);
    return response.toString();
    }


}

module.exports = Fed4Fire;
```

Starting Hyperledger Fabric

2. START FABRIC

Setting up a fabric network involves creating peers and ordered nodes , setting up a Fabric CA, creating a channel (beia), and deploying a chaincode (fed4fire) . All of these steps are defined in the "startfabric.sh" script .

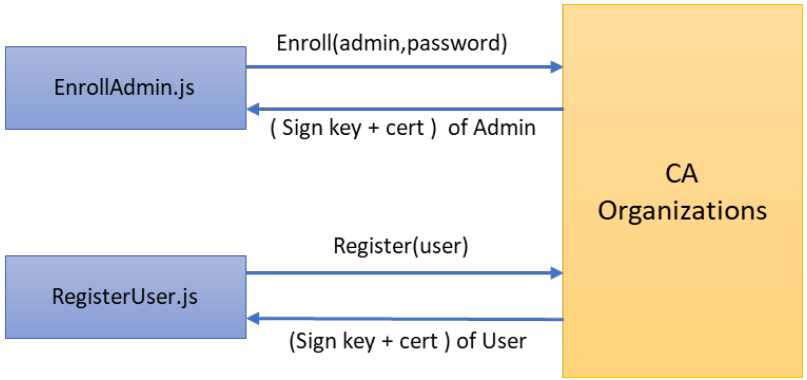
```
CC_SRC_LANGUAGE=${1:-"javascript"}
CC_SRC_PATH="../chaincode/fed4fire/javascript/"

./network.sh up createChannel -c beia -ca -s couchdb
./network.sh deployCC -c beia -ccn fed4fire -ccv 1 -cci initLedger -ccl ${CC_SRC_LANGUAGE} -ccp ${CC_SRC_PATH}
```

```
Creating orderer.example.com    ... done
Creating couchdb1               ... done
Creating couchdb0               ... done
Creating peer0.org1.example.com ... done
Creating peer0.org2.example.com ... done
Creating cli                    ... done
CONTAINER ID   IMAGE                               COMMAND                  CREATED         STATUS          NAMES
a71a1c2e931e   hyperledger/fabric-tools:latest     "/bin/bash"              2 seconds ago   Up Less than a  cli
1adfe02aa9f9   hyperledger/fabric-peer:latest      "peer node start"        6 seconds ago   Up 2 seconds    051/tcp, 7051/tcp, 0.0.0.0:19051->19051/tcp, :::19051->19051/tcp   peer0.org2.example.com
9743bf2a5eec   hyperledger/fabric-peer:latest      "peer node start"        7 seconds ago   Up 2 seconds    051/tcp, 0.0.0.0:17051->17051/tcp, :::17051->17051/tcp              peer0.org1.example.com
80ca4e3badbc   couchdb:3.1.1                       "tini -- /docker-ent…"   9 seconds ago   Up 6 seconds    5984/tcp, :::7984->5984/tcp                                         couchdb1
412b4df52182   couchdb:3.1.1                       "tini -- /docker-ent…"   9 seconds ago   Up 7 seconds    5984/tcp, :::5984->5984/tcp                                         couchdb0
74d39378f889   hyperledger/fabric-orderer:latest   "orderer"                9 seconds ago   Up 6 seconds
```

```
Query chaincode definition successful on peer0.org1 on channel 'beia'
Using organization 2
Querying chaincode definition on peer0.org2 on channel 'beia'...
Attempting to Query committed status on peer0.org2, Retry after 3 seconds.
+ peer lifecycle chaincode querycommitted --channelID beia --name fed4fire
+ res=0
Committed chaincode definition for chaincode 'fed4fire' on channel 'beia':
Version: 1, Sequence: 1, Endorsement Plugin: escc, Validation Plugin: vscc, Approvals: [Org1MSP: true, Org2MSP: true]
Query chaincode definition successful on peer0.org2 on channel 'beia'
Using organization 1
Using organization 2
+ fcn_call='{"function":"initLedger","Args":[]}'
+ infoln 'invoke fcn call:{"function":"initLedger","Args":[]}'
+ println '\033[0;34minvoke fcn call:{"function":"initLedger","Args":[]}\033[0m'
+ echo -e '\033[0;34minvoke fcn call:{"function":"initLedger","Args":[]}\033[0m'
invoke fcn call:{"function":"initLedger","Args":[]}
+ peer chaincode invoke -o localhost:7050 --ordererTLSHostnameOverride orderer.example.com --tls --cafile /home/mandana/fabric/fabric-sam
rganizations/ordererOrganizations/example.com/orderers/orderer.example.com/msp/tlscacerts/tlsca.example.com-cert.pem -C beia -n fed4fire
alhost:7051 --tlsRootCertFiles /home/mandana/fabric/Fabric-Samples/test-network/organizations/peerOrganizations/org1.example.com/peers/pe
m/tls/ca.crt --peerAddresses localhost:9051 --tlsRootCertFiles /home/mandana/fabric/fabric-samples/test-network/organizations/peerOrganiz
.com/peers/peer0.org2.example.com/tls/ca.crt --isInit -c '{"function":"initLedger","Args":[]}'
+ res=0
2022-04-20 07:21:33.330 BST [chaincodeCmd] chaincodeInvokeOrQuery -> INFO 001 Chaincode invoke successful. result: status:200 payload:"su
Invoke transaction successful on peer0.org1 peer0.org2 on channel 'beia'
```

Register And Enroll in Fabric CA

The first step is to enroll the administrator in Fabric CA . This will generate the admin user's signed certificate and private key stored in the wallet/admin directory. This key-pair is used to register and enroll other users in the organization and obtain the private key and user's signed certificate . This data is stored in the wallet/user directory and will be used for the interaction in the chaincode .



Register And Enroll in Fabric CA

1. ENROLL ADMIN

```
const enrollment = await ca.enroll({ enrollmentID: 'admin', enrollmentSecret: 'adminpw' });
const x509Identity = {
    credentials: {
        certificate: enrollment.certificate,
        privateKey: enrollment.key.toBytes(),
    },
    mspId: 'Org1MSP',
    type: 'X.509',
};
await wallet.put('admin', x509Identity);
console.log('Successfully enrolled admin user "admin" and
```

```
yfile:/etc/hyperledger/fabric-ca-server/msp/keystore/IssuerSecretKey RevocationPublicKeyfile:/etc/hyperledger/fabric-ca-server/IssuerR
ationPublicKey RevocationPrivateKeyfile:/etc/hyperledger/fabric-ca-server/msp/keystore/IssuerRevocationPrivateKey RHPoolSize:100 Nonce
ration:15s NonceSweepInterval:15m}}
        ca_org1|2022/04/20 06:42:43 [DEBUG] DB: Getting identity admin
        ca_org1|2022/04/20 06:42:43 [DEBUG] DB: Login user admin with max enrollments of -1 and state of 1
        ca_org1|2022/04/20 06:42:43 [DEBUG] DB: identity admin successfully logged in
        ca_org1|2022/04/20 06:42:43 [DEBUG] DB: Getting identity admin
        ca_org1|2022/04/20 06:42:43 [DEBUG] Processing sign request: id=admin, CommonName=admin, Subject=<nil>
        ca_org1|2022/04/20 06:42:43 [DEBUG] Request is not for a CA signing certificate
        ca_org1|2022/04/20 06:42:43 [DEBUG] Checking CSR fields to make sure that they do not exceed maximum character limits
        ca_org1|2022/04/20 06:42:43 [DEBUG] Finished processing sign request
        ca_org1|2022/04/20 06:42:43 [DEBUG] DB: Getting identity admin
        ca_org1|2022/04/20 06:42:43 [INFO] signed certificate with serial number 115798317097029811426362466240179315524159932629
        ca_org1|2022/04/20 06:42:43 [DEBUG] DB: Insert Certificate
        ca_org1|2022/04/20 06:42:43 [DEBUG] Saved serial number as hex 1448937cf4a9d00faa1f9633a5e4d54b8f4408d5
        ca_org1|2022/04/20 06:42:43 [DEBUG] saved certificate with serial number 115798317097029811426362466240179315524159932629
        ca_org1|2022/04/20 06:42:43 [DEBUG] Successfully incremented state for identity admin to 2
        ca_org1|2022/04/20 06:42:43 [INFO] 172.23.0.1:45660 POST /api/v1/enroll 201 0 "OK"
```

```
mandana@hyperledger-vm:~/fabric/fabric-samples/beia/javascript$ cat ./wallet/admin.id
{"credentials":{"certificate":"-----BEGIN CERTIFICATE-----\nMIIB8zCCAZmgAwIBAgIUFEiTfPSp0A+qH5YzpeTVS49ECNUwCgYIKoZIzj0EAwIw\ncDELMAkGA1UE
BhMCVVMxFzAVBgNVBAgTDk5vcnRoIENhcm9saW5hMQ8wDQYDVQQH\nEwZEdXJoYW0xGTAXBgNVBAoTEG9yZzEuZXhhbXBsZS5jb20xHDAaBgNVBAMTE2Nh\nLm9yZzEuZXhhbXBsZS
5jb20wHhcNMjIwNDIwMDYzODAwWhcNMjMwNDIwMDY0MzAw\nWjAhMQ8wDQYDVQQLEwZjbGllbnQxDjAMBgNVBAMTBWFkbWluMFkwEwYHKoZIzj0C\nAQYIKoZIzj0DAQcDQgAEP3iK
4krNePzXke+l43eefHfhL/V/JBjvhLXR0VJYv/8P\nVHfgvPb1hm8JKNPV1h1O0njkyQeDwUddGGI/wTJOcqNgMF4wDgYDVR0PAQH/BAQD\nAgeAMAwGA1UdEwEB/wQCMAAwHQYDVR
0OBBYEFH4CztGnl3B3d0Ts6+z7oVUs3aQX\nMB8GA1UdIwQYMBaAFHrYrQhlQHuB8KGUAi+yGYNbZylZMAoGCCqGSM49BAMCA0gA\nMEUCIQD+j+QFqjBLpSgXXUaX+yjoDe87qYAA
KzccQo1alGqFCQIgQYXn9FJ5ykuC\niqUcIrbxmcz7f0JENFcC82WPioj2ayo=\n-----END CERTIFICATE-----\n","privateKey":"-----BEGIN PRIVATE KEY-----\r\n
MIGHAgEAMBMGByqGSM49AgEGCCqGSM49AwEHBG0wawIBAQQgIQlRNSfwzLzknWhA\r\nP06pVGjqcDYgr1cwzBleZUe/DjChRANCAAQ/eIriSs14/NeR76Xjd558d+Ev9X8k\r\nGO
+EtdHRUli//w9Ud+C89vWGbwko09XWHU7SeOTJB4PBR10YYj/BMk5y\r\n-----END PRIVATE KEY-----\r\n"},"mspId":"Org1MSP","type":"X.509","version":1}man
```

Register And Enroll in Fabric CA

1. REGISTER USER (USER01)

```
const secret = await ca.register({
    affiliation: 'org1.department1',
    enrollmentID: 'User01',
    role: 'client'
}, adminUser);
const enrollment = await ca.enroll({
    enrollmentID: 'User01',
    enrollmentSecret: secret
});
const x509Identity = {
    credentials: {
        certificate: enrollment.certificate,
        privateKey: enrollment.key.toBytes(),
    },
    mspId: 'Org1MSP',
    type: 'X.509',
};
await wallet.put('User01', x509Identity);
console.log('Successfully registered and enrolled admin user "User01"
```

```
ca_org1|2022/04/20 06:56:04 [INFO] signed certificate with serial number 434125838614449416270159607079630761028896975397
ca_org1|2022/04/20 06:56:04 [DEBUG] DB: Insert Certificate
ca_org1|2022/04/20 06:56:04 [DEBUG] Saved serial number as hex 4c0ae03917dade7f6b26c0f9e5f85fd18dcc6a25
ca_org1|2022/04/20 06:56:04 [DEBUG] saved certificate with serial number 434125838614449416270159607079630761028896975397
ca_org1|2022/04/20 06:56:04 [DEBUG] Successfully incremented state for identity User01 to 1
ca_org1|2022/04/20 06:56:04 [INFO] 172.23.0.1:45664 POST /api/v1/enroll 201 0 "OK"
```

```
mandana@hyperledger-vm:~/fabric/fabric-samples/beia/javascript$ cat ./wallet/User01.id
{"credentials":{"certificate":"-----BEGIN CERTIFICATE-----\nMIICgjCCAiigAwIBAgIUTArgORfa3n9rJsD55fhf0Y3MaiUwCgYIKoZIzj0EAwI
BhMCVVMxFzAVBgNVBAgTDk5vcnRoIENhcm9saW5hMQ8wDQYDVQQH\nEwZEdXJoYW0xGTAXBgNVBAoTEG9yZzEuZXhhbXBsZS5jb20xHDAaBgNVBAMTE2Nh\nLmS
5jb20wHhcNMjIwNDIwMDY1MTAwWhcNMjMwNDIwMDY1NjAw\nWjBDMTAwDQYDVQQLEwZjbGllbnQwCwYDVQQLEwRvcmcxMBIGA1UECxMLZGVwYXJ0\nbWVudDExI
ZXIwMTBZMBMGByqGSM49AgEGCCqGSM49AwEHA0IA\nBLrrzCk+7/xPG+3/8vgucsoiui6PIOKEqqzhvF46pplX7h3HPNExDvAGIvrmg66V\naqhbWRx5jwuethy
kwDgYDVR0PAQH/BAQDAgeAMAwGA1UdEwEB\n/wQCMAAwHQYDVR0OBBYEFK/uAYOxjaEbpy3REzDJ1SqnI/RvMB8GA1UdIwQYMBaA\nFHrYrQhlQHuB8KGUAi+yC
BAUGBwgBBF17ImF0dHJzIjp7Imhm\nLkFmZmlsaWF0aW9uIjoib3JnMS5kZXBhcnRtZW50MSIsImhmLkVucm9sbG1lbnRJ\nRCI6IlVzZXIwMSIsImhmLlR5cGU
0wCgYIKoZIzj0EAwIDSAAw\nRQIhAK4froi5GvpZr5OgOSI0DRv1pHUzUgAMsBMOmzJrYEODAiABZJl9bl9yN1FR\nMk2DdcNF+IIZVMuHOwT6NC0Xa+qI9Q==\
FICATE-----\n","privateKey":"-----BEGIN PRIVATE KEY-----\r\nMIGHAgEAMBMGByqGSM49AgEGCCqGSM49AwEHBG0wawIBAQQgh8UQveSjaakapkF
QWPgIhkAMOaq89sjWhRANCAAS668wpPu/8Txvt//L4LnLKIroujyDi\r\nhKqs4bxeOqaZV+4dxzzRMQ7wBiL65oOulWqoW1kceY8LnrYck5MPQtlz\r\n-----
-----\r\n"},"mspId":"Org1MSP","type":"X.509","version":1}mandana@hyperledger-vm:~/fabric/fabric-samples/beia/javascript$
```

Writing on Hyperledger Fabric

In this section, programs (invoke.js and query.js) are defined to call the “fed4fire" chaincode of the “beia" channel so that registered users can submit data to Hyperledger and evaluate their transaction.

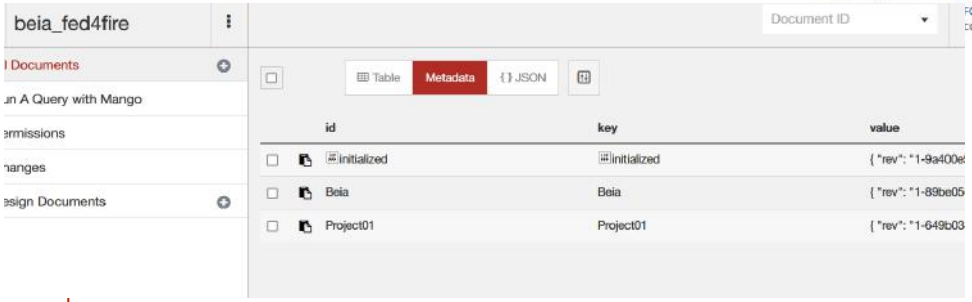
User01 submits the transaction (Project01 , Value01) in Hyperledger ,then the user evaluates the transaction and checks the result .

Writing on Hyperledger Fabric

1. SUBMIT TRANSACTION

```
// Check to see if we've already enrolled the user.
const identity = await wallet.get('User01');
if (!identity) {
    console.log('An identity for the user "User01" does not exist in the wallet');
    console.log('Run the registerUser.js application before retrying');
    return;
}

const gateway = new Gateway();
await gateway.connect(ccp, { wallet, identity: 'User01', discovery: { enabled: true, asLocalhost: true } });

const network = await gateway.getNetwork('beia');

const contract = network.getContract('fed4fire');

await contract.submitTransaction("writeData", "Project01", "Value01");
console.log('Transaction has been submitted');

await gateway.disconnect();
```

```
mandana@hyperledger-vm:~/fabric/fabric-samples/beia/javascript$ node invoke.js
Wallet path: /home/mandana/fabric/fabric-samples/beia/javascript/wallet
Transaction has been submitted
                                                                                couchdb1|[notice] 2022-04-20T07:11:59.380481
Z nonode@nohost <0.27461.0> 15adfbcd9c couchdb1:5984 172.23.0.9 admin POST /beia_fed4fire/_bulk_docs 201 ok 89
                                                                                couchdb1|[notice] 2022-04-20T07:11:59.396113
Z nonode@nohost <0.27490.0> 7ce47c881d couchdb1:5984 172.23.0.9 admin GET /beia_fed4fire/_index 200 ok 10
                                                                                couchdb0|[notice] 2022-04-20T07:11:59.397506
Z nonode@nohost <0.27506.0> b86346bcc8 couchdb0:5984 172.23.0.8 admin PUT /beia_/statedb_savepoint 201 ok 37
                                                                                peer0.org1.example.com|2022-04-20 07:11:59.458 UTC [kvledg
er] CommitLegacy -> INFO 07d [beia] Committed block [7] with 1 transaction(s) in 305ms (state_validation=0ms block_and_pvtdata_commit=118m
s state_commit=129ms) commitHash=[7ae3ca0f68aaf9de794db1a5a32b46984329d95b438cb796275f221ea632b4f5]
                                                                                couchdb1|[notice] 2022-04-20T07:11:59.462977
Z nonode@nohost <0.27461.0> f054e7fe18 couchdb1:5984 172.23.0.9 admin GET /beia_/statedb_savepoint?attachments=true 200 ok 80
                                                                                peer0.org1.example.com|2022-04-20 07:11:59.467 UTC [comm.g
rpc.server] 1 -> INFO 07e streaming call completed grpc.service=protos.Deliver grpc.method=DeliverFiltered grpc.peer_address=172.23.0.1:36
112 grpc.peer_subject="CN=fabric-common" error="context finished before block retrieved: context canceled" grpc.code=Unknown grpc.call_dur
ation=2.443388483s
                                                                                couchdb1|[notice] 2022-04-20T07:11:59.498329
Z nonode@nohost <0.27461.0> 178bbe42fa couchdb1:5984 172.23.0.9 admin PUT /beia_/statedb_savepoint 201 ok 19
                                                                                peer0.org2.example.com|2022-04-20 07:11:59.527 UTC [kvledg
FO 076 [beia] Committed block [7] with 1 transaction(s) in 371ms (state_validation=0ms block_and_pvtdata_commit=128m
commitHash=[7ae3ca0f68aaf9de794db1a5a32b46984329d95b438cb796275f221ea632b4f5]
```



Writing on Hyperledger Fabric

2. EVALUATE TRANSACTION

```
    const gateway = new Gateway();
    await gateway.connect(ccp, { wallet, identity: 'User01', discovery: { enabled: true, asLocalhost: true } });

    const network = await gateway.getNetwork('beia');

    const contract = network.getContract('fed4fire');

    const result = await contract.evaluateTransaction('readData','Project01');
    console.log(`Transaction has been evaluated, result is: ${result.toString()}`);

    // Disconnect from the gateway.
    await gateway.disconnect();

} catch (error) {
```

```
mandana@hyperledger-vm:~/fabric/fabric-samples/beia/javascript$ node query.js
Wallet path: /home/mandana/fabric/fabric-samples/beia/javascript/wallet
Transaction has been evaluated, result is: Value01
mandana@hyperledger-vm:~/fabric/fabric-samples/beia/javascript$
```

```
rpc.server] 1 -> INFO 07f unary call completed grpc.service=discovery.Discovery grpc.method=Discover grpc.peer_address=172.23.0.1:36114 gr
pc.peer_subject="CN=fabric-common" grpc.code=OK grpc.call_duration=597.864µs
                                                                                couchdb0|[notice] 2022-04-20T07:25:49.918106
Z nonode@nohost <0.1765.1> 9412f4e999 couchdb0:5984 172.23.0.8 admin GET /beia_fed4fire/Project01?attachments=true 200 ok 10
dev-peer0.org1.example.com-fed4fire_1-38ad64274e36171ba00fad5e170111e5565a164f6db583d4a9008d35c6e58e76|2022-04-20T07:25:49.921Z info [c-ap
i:lib/handler.js]                       [beia-f7d8201b] Calling chaincode Invoke() succeeded. Sending COMPLETED message back to pee
r
                                                        peer0.org1.example.com|2022-04-20 07:25:49.925 UTC [endors
er] callChaincode -> INFO 080 finished chaincode: fed4fire duration: 25ms channel=beia txID=f7d8201b
                                                        peer0.org1.example.com|2022-04-20 07:25:49.926 UTC [comm.g
rpc.server] 1 -> INFO 081 unary call completed grpc.service=protos.Endorser grpc.method=ProcessProposal grpc.peer_address=172.23.0.1:36116
 grpc.peer_subject="CN=fabric-common" grpc.code=OK grpc.call_duration=26.495018ms
```

Conclusions

The main experimental results of the SO-SHARED project are:

- The development and implementation of product functionalities in an agile fashion to ensure the right direction of the project early on;
- Final testing process, looking at functional and non-functional aspects to guarantee readiness of the solution and scalability.

Business Impact

- We will further design and implement innovative business models that, based on blockchain technologies, will provide traceability, transparency, trust and gamification features, such as securely tracking the sensor data information as well as the identity of the stakeholders and rewarding them with coins/tokens for their behavior.
- Blockchain is contributing to the evolution of decentralized autonomous businesses by promoting disruptive cooperation procedures and process modelling between heterogeneous stakeholders.

Feedback

USED RESOURCES – CLOUD COMPUTING TESTBED (EXOGENI)

- Large amount of available resources in the FED4FIRE+ consortium
- Setup – the deployment of the experiment was a liitle bit hard because UvA (Exogeni resource) was decomissioned and we quite needed a bit more of resources
- The funding helped us to finalise the SO-SHARED solution we were working on

This project has received funding from the European Union's Horizon 2020 research and innovation programme, which is co-funded by the European Commission and the Swiss State Secretariat for Education, Research and Innovation, under grant agreement No 732638.

Contact person:

Dr. Eng. George Suciu Jr.
BEIA Consult International
george@beia.ro / Twitter: @GeorgeSuciuG
www.beiaro.eu
www.beia-telemetrie.ro

WWW.FED4FIRE.EU