How it works documentation.

This document is a simple guide to demonstrate how to start the application using the hyperledger fabric test network.

Step 1: Start Hyperledger Fabric Testnet and Add Organization 3 to the Channel

The Application utilized the testnet provided by Fabric Sample from Hyperledger Fabric GitHub repository. Please follow the guide from the official document to start the network before installing the chaincode.
https://github.com/hyperledger/fabric-samples

1. Navigate to the Fabric Samples repository.
2. Follow the instructions provided in the official documentation to start the Hyperledger Fabric testnet.
3. Add Organization 3 to the channel.

Step 2: Clone the Application and Chaincode Repository

```
git clone https://github.com/LoChingHei/Capstone_YachtTourism.git
```

Step 3: Install Chaincode on Peer Nodes

Example of installing chaincode to peer node 1:

```
export CORE_PEER_TLS_ENABLED=true
export CORE_PEER_LOCALMSPID="Org1MSP"
export CORE_PEER_TLS_ROOTCERT_FILE=${PWD}/organizations/peerOrganizations/org1.example.com/peers/peer0.org1.example.com/tls/ca.crt
export CORE_PEER_MSPCONFIGPATH=${PWD}/organizations/peerOrganizations/org1.example.com/users/Admin@org1.example.com/msp
export CORE_PEER_ADDRESS=localhost:7051

peer lifecycle chaincode install yacht.tar.gz
```

Step 4: Approve Chaincode (At Least 2 Organizations Need to Approve)

Example of approving chaincode to peer node 1:

```
“
export CORE_PEER_LOCALMSPID="Org1MSP"
export CORE_PEER_MSPCONFIGPATH=${PWD}/organizations/peerOrganizations/org1.example.com/users/Admin@org1.example.com/msp
export CORE_PEER_TLS_ROOTCERT_FILE=${PWD}/organizations/peerOrganizations/org1.example.com/peers/peer0.org1.example.com/tls/ca.crt
export CORE_PEER_ADDRESS=localhost:7051

peer lifecycle chaincode approveformyorg -o localhost:7050 --ordererTLSHostnameOverride orderer.example.com --channelID mychannel --name yacht --version 1.0 --package-id $CC_PACKAGE_ID --sequence 1 --tls --cafile "${PWD}/organizations/ordererOrganizations/example.com/orderers/orderer.example.com/msp/tlscacerts/tlsca.example.com-cert.pem"
“””
```

Step 5: Check Chaincode Readiness

```
“””
peer lifecycle chaincode checkcommitreadiness --channelID mychannel --name yacht --version 1.0 --sequence 1 --tls --cafile "${PWD}/organizations/ordererOrganizations/example.com/orderers/orderer.example.com/msp/tlscacerts/tlsca.example.com-cert.pem" --output json
“””
```

Step 6: Commit the Chaincode

```
peer lifecycle chaincode commit -o localhost:7050 --ordererTLSHostnameOverride orderer.example.com --channelID mychannel --name yacht --version 1.0 --sequence 1 --tls --cafile "${PWD}/organizations/ordererOrganizations/example.com/orderers/orderer.example.com/msp/tlscacerts/tlsca.example.com-cert.pem" --peerAddresses localhost:7051 --tlsRootCertFiles
```

```
"${PWD}/organizations/peerOrganizations/org1.example.com/peers/peer0
.org1.example.com/tls/ca.crt" --peerAddresses localhost:9051
--tlsRootCertFiles
"${PWD}/organizations/peerOrganizations/org2.example.com/peers/peer0
.org2.example.com/tls/ca.crt" --peerAddresses localhost:11051
--tlsRootCertFiles
"${PWD}/organizations/peerOrganizations/org3.example.com/peers/peer0
.org3.example.com/tls/ca.crt"
```

The chaincode is now ready to interact with the network for query and invoke!

Step 7: Start the Application

Enroller and register users in different organizations.

```
cd yacht-application
npm install

node enrollUser.js 'CAAdmin@org1.example.com' admin adminpw

node registerUser.js 'CAAdmin@org1.example.com'
'User1@org1.example.com' '{"secret": "userpw"}'

node enrollUser.js 'User1@org1.example.com' 'User1@org1.example.com'
userpw

node enrollUser.js 'CAAdmin@org2.example.com' admin adminpw

node registerUser.js 'CAAdmin@org2.example.com'
'User1@org2.example.com' '{"secret": "userpw"}'

node enrollUser.js 'User1@org2.example.com' 'User1@org2.example.com'
userpw

node enrollUser.js 'CAAdmin@org3.example.com' admin adminpw

node registerUser.js 'CAAdmin@org3.example.com'
'User1@org3.example.com' '{"secret": "userpw"}'

node enrollUser.js 'User1@org3.example.com' 'User1@org3.example.com'
userpw
```

Step 8: Start the Application Backend

```
node server.js
```

Step 9: Start the UI

```
npm start
```

For more information, you can refer to our Report.