

INTRODUCTION

A Letter of Credit is an arrangement whereby Bank acting at the request of a customer (Importer / Buyer), undertakes to pay for the goods / services, to a third party (Exporter / Beneficiary) by a given date, on documents being presented in compliance with the conditions laid down.

In this **use case of Letter of Credit**, has two parties' seller and buyer undertaking goods exchange in the presences of Bank. This document is provided to serve as an HLF Java SDK integration blockchain network on IBM cloud and test the smart contracts of the use case Letter of Credit.

TECHNOLOGY USED

The tools and software required to set up the use case are

- IBM cloud
- VS Code with Go Lang and IBM Blockchain extension
- Eclipse IDE with Java and maven installed.

STEPS

STEP 1: IBM BLOCKCHAIN NETWORK SETUP

In the IBM cloud, the following steps are performed to setup the blockchain platform:

- Create “Kubernetes Service”
- In Kubernetes Service Create a Cluster
- Create a Blockchain Platform and Link with the created Kubernetes cluster
- Launch the IBM Blockchain Platform Console
- Create Certificate Authority
- Once Certificate Authority is created then Create Associate Identity
- Register all the required User with the help of Register User by providing Enroll ID, Enroll Secret and Type
- In Organization, Create MSP Definition, with MSP display name and MSP ID of the selected Certificate Authority for the required organization and orderers
- Create Peers for the required organization and orderers
- Add Organization to Consortium Members
- Create Channels based on the requirement and join peers to the channel
- Install Smart contract and propose it
- Organizations will approve and commit the installed Smart Contract

STEP 2: SMART CONTRACT

Based on the use case’s requirement, the smart contract/ chain code is programmed in Go Lang with the help of VS code.

Required Structures are defined with the required properties. This use case includes below mentioned structures i.e LetterOfCredit, Transaction and QueryResult.

LetterOfCredit		
Buyer	String	PartyA
dateOfCreation	String	Loc creation date by Bank
BuyerBank	String	PartyA’s Bank
Seller	String	Party B
Remarks	String	Remarks if any
locStatus	String	Accepted/Reject by PartyA
IsShipGoodsBySeller	String	Yes/No
IsGoodsConfirmByBuyer	String	Yes/No
IsVerifiedSellerDocs	String	Yes/No
IsVerifiedBuyerDocs	String	Yes/No
ReleasePayment	String	Yes/No
PaymentAmount	String	Amount paid by PartyA Bank
Transaction	Transaction[]	Transaction Details

Transaction		
TransactionId	String	UniqueId
TransactionType	String	AddLoc/AcceptLoc/RejectLoc/ShipGoods/VerifiedGoods ByBuyer/VerifiedSellerDocsByBank/REleasePayemtn
TransactionTimestamp	String	Timestamp
TransactionInitiatedBy	String	Who initiate Transaction
Remarks	String	Remark if any
LetterOfCreditNo	String	LetterOfCreditNo

Query Result		
LetterOfCreditNo	String	UniqueId LOC Id
Record	* LetterOfCredit	LetterOfCredit Details

The Smart Contract include below mentioned API,

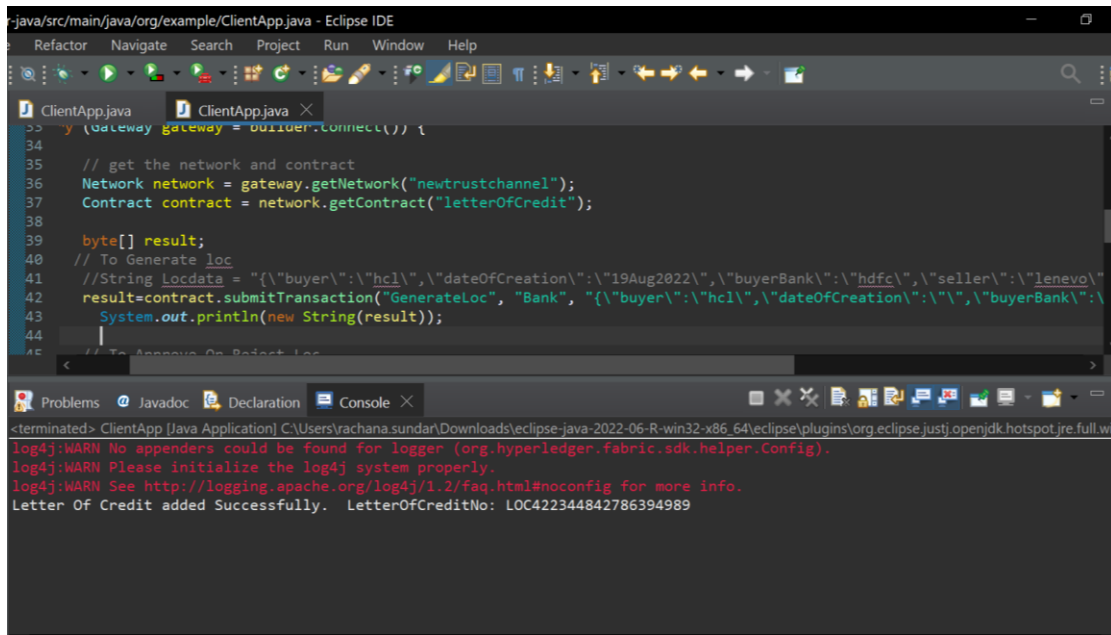
API Name		
GenerateLoc	Participant-Bank	Bank generates LOC and LOC number
ApproveOrRejectLoC	Pariticipant-ParyB	Seller will Approve Or Reject the generated LOC
ShipGoodsBySeller	Participant - PartyB	Verify the Goods Shipment Status by the seller
GoodsVerifiedBySeller	Participant - PartyB	Verify the Goods by the Seller
GoodsVerifiedByBuyer	Participant - PartyA	Verify the Goods by the Buyer
VerifiedShippingGoodsByBank	Pariticipant - BankA	Verify the Goods Status by the Bank
ReleasePayment	Participant -Bank	Pay Amount if verified by Bank
GetLoc		Returns the LOC data of a particular LOC number
QueryAllLoc		Returns All LOC stored

STEP 3: JAVA CLIENT

- Java client created with the required dependencies
- Enroll the admin user, and import the new identity into the wallet.
- Register the user, enroll the user, and import the new identity into the wallet.
- Run the Client Application which includes the submit Transaction of all the API mentioned in the chain code.

OUTPUT:

- GenerateLoc:

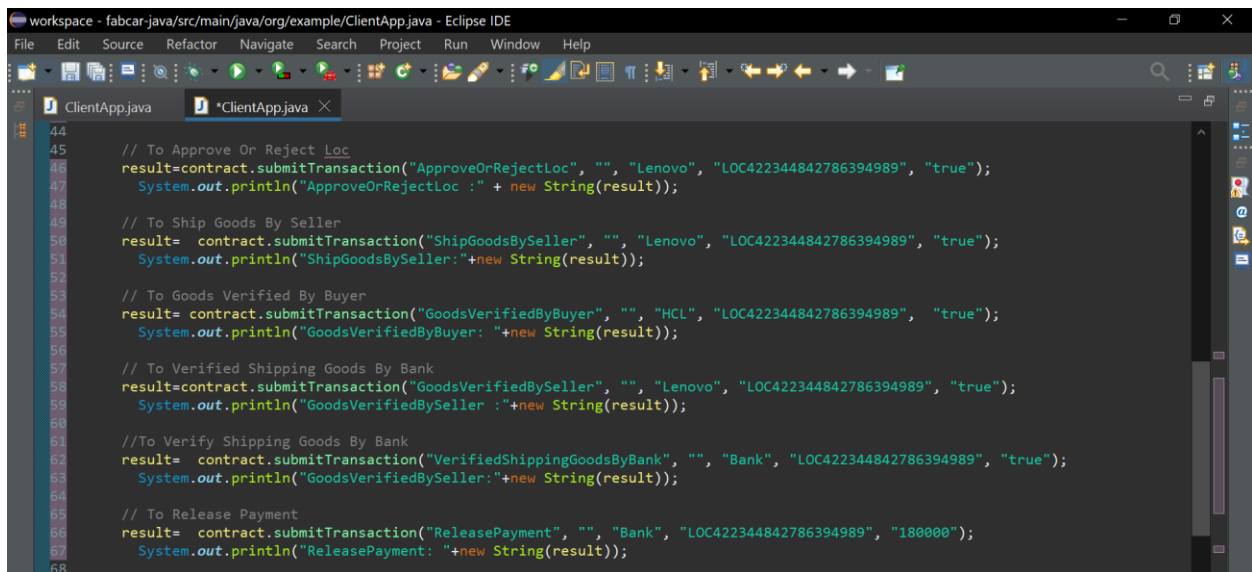


```
java/src/main/java/org/example/ClientApp.java - Eclipse IDE
Refactor  Navigate  Search  Project  Run  Window  Help

ClientApp.java
35 // Gateway gateway = builder.connect();
36 // get the network and contract
37 Network network = gateway.getNetwork("newtrustchannel");
38 Contract contract = network.getContract("letterOfCredit");
39
40 byte[] result;
41 // To Generate Loc
42 //String Locdata = "{\"buyer\":\"hcl\",\"dateOfCreation\":\"19Aug2022\",\"buyerBank\":\"hdfc\",\"seller\":\"lenevo\"";
43 result=contract.submitTransaction("GenerateLoc", "Bank", "{\"buyer\":\"hcl\",\"dateOfCreation\":\"\",\"buyerBank\":\"");
44 System.out.println(new String(result));
45
46 // To Approve Or Reject Loc

<terminated> ClientApp [Java Application] C:\Users\vachana.sundar\Downloads\eclipse-java-2022-06-R-win32-x86_64\eclipse\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win
log4j:WARN No appenders could be found for logger (org.hyperledger.fabric.sdk.helper.Config).
log4j:WARN Please initialize the log4j system properly.
log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more info.
Letter Of Credit added Successfully. LetterOfCreditNo: LOC422344842786394989
```

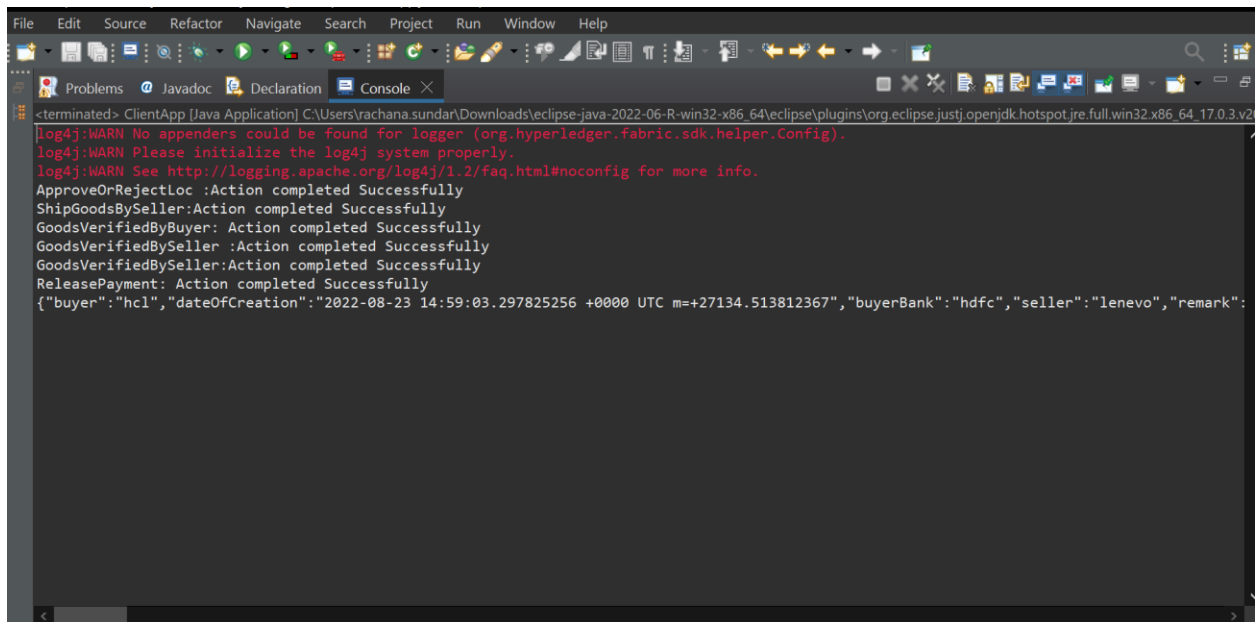
- Using the LetterOfCreditNo generated and running other APIs



```
workspace - fabcar-java/src/main/java/org/example/ClientApp.java - Eclipse IDE
File  Edit  Source  Refactor  Navigate  Search  Project  Run  Window  Help

ClientApp.java
44
45 // To Approve Or Reject Loc
46 result=contract.submitTransaction("ApproveOrRejectLoc", "", "Lenovo", "LOC422344842786394989", "true");
47 System.out.println("ApproveOrRejectLoc : " + new String(result));
48
49 // To Ship Goods By Seller
50 result= contract.submitTransaction("ShipGoodsBySeller", "", "Lenovo", "LOC422344842786394989", "true");
51 System.out.println("ShipGoodsBySeller:"+new String(result));
52
53 // To Goods Verified By Buyer
54 result= contract.submitTransaction("GoodsVerifiedByBuyer", "", "HCL", "LOC422344842786394989", "true");
55 System.out.println("GoodsVerifiedByBuyer: "+new String(result));
56
57 // To Verified Shipping Goods By Bank
58 result=contract.submitTransaction("GoodsVerifiedBySeller", "", "Lenovo", "LOC422344842786394989", "true");
59 System.out.println("GoodsVerifiedBySeller :"+new String(result));
60
61 //To Verify Shipping Goods By Bank
62 result= contract.submitTransaction("VerifiedShippingGoodsByBank", "", "Bank", "LOC422344842786394989", "true");
63 System.out.println("GoodsVerifiedBySeller:"+new String(result));
64
65 // To Release Payment
66 result= contract.submitTransaction("ReleasePayment", "", "Bank", "LOC422344842786394989", "180000");
67 System.out.println("ReleasePayment: "+new String(result));
68
```

Output of APIs:



```
<terminated> ClientApp [Java Application] C:\Users\rachana.sundar\Downloads\eclipse-java-2022-06-R-win32-x86_64\eclipse\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.17.0.3.v2
log4j:WARN No appenders could be found for logger (org.hyperledger.fabric.sdk.helper.Config).
log4j:WARN Please initialize the log4j system properly.
log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more info.
ApproveOrRejectLoc :Action completed Successfully
ShipGoodsBySeller:Action completed Successfully
GoodsVerifiedByBuyer: Action completed Successfully
GoodsVerifiedBySeller:Action completed Successfully
GoodsVerifiedBySeller:Action completed Successfully
ReleasePayment: Action completed Successfully
{"buyer":"hcl","dateOfCreation":"2022-08-23 14:59:03.297825256 +0000 UTC m=+27134.513812367","buyerBank":"hdfc","seller":"lenevo","remark":
```

```
{ "buyer": "hcl", "dateOfCreation": "2022-08-23 14:59:03.297825256 +0000 UTC
m=+27134.513812367", "buyerBank": "hdfc", "seller": "lenevo", "remark": "Ok", "locStatus": "Accepted", "is
ShipGoodsBySeller": "Yes", "isVerifiedSellerDocs": "Yes", "isVerifiedBuyerDocs": "Yes", "isVerifiedShipp
ingGoodsByBank": "Yes", "releasePayment": "Yes", "paymentAmount": "180000", "transaction": [{ "transacti
onId": "TX422344842819949421", "transactionType": "GenerateLoc", "timeStamp": "2022-08-23
14:59:03.298024705 +0000 UTC
m=+27134.514011764", "transactionInitiatedBy": "Bank", "remarks": "LOC Generated
", "letterOfCreditNo": "LOC422344842786394989"}, {" "transactionId": "TX422345208194159469", "transa
ctionType": "ApproveOrRejectLoc", "timeStamp": "2022-08-23 15:02:41.077679707 +0000 UTC
m=+27352.293666615", "transactionInitiatedBy": "Lenovo", "remarks": "Accepted the Letter of
Credit", "letterOfCreditNo": "LOC422344842786394989"}, {" "transactionId": "TX422345214804382570", "t
ransactionType": "ShipGoodsBySeller", "timeStamp": "2022-08-23 15:02:45.015478925 +0000 UTC
m=+27357.275666707", "transactionInitiatedBy": "Lenovo", "remarks": "Goods shipped by the
Seller", "letterOfCreditNo": "LOC422344842786394989"}, {" "transactionId": "TX422345219971765098", "t
ransactionType": "GoodsVerifiedByBuyer", "timeStamp": "2022-08-23 15:02:48.094536128 +0000 UTC
m=+27360.354724652", "transactionInitiatedBy": "HCL", "remarks": "Goods verified by the
Buyer", "letterOfCreditNo": "LOC422344842786394989"}, {" "transactionId": "TX422345225155924845", "t
ransactionType": "GoodsVerifiedBySeller", "timeStamp": "2022-08-23 15:02:51.188440532 +0000 UTC
m=+27362.404428509", "transactionInitiatedBy": "Lenovo", "remarks": "Goods verified by the
Buyer", "letterOfCreditNo": "LOC422344842786394989"}, {" "transactionId": "TX422345230390416237", "t
ransactionType": "VerifiedShippingGoodsByBank", "timeStamp": "2022-08-23 15:02:54.302608589
+0000 UTC m=+27365.518596601", "transactionInitiatedBy": "Bank", "remarks": "Goods shipment
verified by the
Bank", "letterOfCreditNo": "LOC422344842786394989"}, {" "transactionId": "TX422345235524244330", "tr
ansactionType": "ReleasePayment", "timeStamp": "2022-08-23 15:02:57.368495005 +0000 UTC
m=+27369.628729618", "transactionInitiatedBy": "Bank", "remarks": "Amount paid
successfully", "letterOfCreditNo": "LOC422344842786394989"} ] }
```

```

50 // result= contract.submitTransaction("ShipGoodsBySeller", "", "Lenovo", "", "true");
51 // System.out.println("ShipGoodsBySeller:"+new String(result));
52 //
53 // // To Goods Verified By Buyer
54 // result= contract.submitTransaction("GoodsVerifiedByBuyer", "", "HCL", "", "true");
55 // System.out.println("GoodsVerifiedByBuyer: "+new String(result));
56 //
57 // // To Verified Shipping Goods By Bank
58 // result=contract.submitTransaction("GoodsVerifiedBySeller", "", "Lenovo", "", "true");
59 // System.out.println("GoodsVerifiedBySeller :"+new String(result));
60 //
61 // //To Verify Shipping Goods By Bank
62 // result= contract.submitTransaction("VerifiedShippingGoodsByBank", "", "Bank", "", "true");
63 // System.out.println("VerifiedShippingGoodsByBank:"+new String(result));
64 //
65 // // To Release Payment
66 // result= contract.submitTransaction("ReleasePayment", "", "Bank", "LOC422344842786394989", "180000");
67 // System.out.println("ReleasePayment: "+new String(result));
68 //
69 // //Querying to get loc by ID
70 // result = contract.evaluateTransaction("GetLoc", "LOC422344842786394989");
71 // System.out.println(new String(result));
72 // }
73 // }
74 //
75 // }
76 //

```

```

{"buyer":"hcl","dateOfCreation":"2022-08-23 14:59:03.297825256 +0000 UTC
m=+27134.513812367","buyerBank":"hdfc","seller":"lenevo","remark":"Ok","locStatus":"Accepted","is
ShipGoodsBySeller":"Yes","isVerifiedSellerDocs":"Yes","isVerifiedBuyerDocs":"Yes","isVerifiedShipp
ingGoodsByBank":"Yes","releasePayment":"Yes","paymentAmount":"180000","transaction":[{"transacti
onId":"TX422344842819949421","transactionType":"GenerateLoc","timeStamp":"2022-08-23
14:59:03.298024705 +0000 UTC
m=+27134.514011764","transactionInitiatedBy":"Bank","remarks":"LOC Generated
","letterOfCreditNo":"LOC422344842786394989"},{"transactionId":"TX422345208194159469","transa
ctionType":"ApproveOrRejectLoc","timeStamp":"2022-08-23 15:02:41.077679707 +0000 UTC
m=+27352.293666615","transactionInitiatedBy":"Lenovo","remarks":"Accepted the Letter of
Credit","letterOfCreditNo":"LOC422344842786394989"},{"transactionId":"TX422345214804382570","t
ransactionType":"ShipGoodsBySeller","timeStamp":"2022-08-23 15:02:45.015478925 +0000 UTC
m=+27357.275666707","transactionInitiatedBy":"Lenovo","remarks":"Goods shipped by the
Seller","letterOfCreditNo":"LOC422344842786394989"},{"transactionId":"TX422345219971765098","t
ransactionType":"GoodsVerifiedByBuyer","timeStamp":"2022-08-23 15:02:48.094536128 +0000 UTC
m=+27360.354724652","transactionInitiatedBy":"HCL","remarks":"Goods verified by the
Buyer","letterOfCreditNo":"LOC422344842786394989"},{"transactionId":"TX422345225155924845","t
ransactionType":"GoodsVerifiedBySeller","timeStamp":"2022-08-23 15:02:51.188440532 +0000 UTC
m=+27362.404428509","transactionInitiatedBy":"Lenovo","remarks":"Goods verified by the
Buyer","letterOfCreditNo":"LOC422344842786394989"},{"transactionId":"TX422345230390416237","t
ransactionType":"VerifiedShippingGoodsByBank","timeStamp":"2022-08-23 15:02:54.302608589
+0000 UTC m=+27365.518596601","transactionInitiatedBy":"Bank","remarks":"Goods shipment
verified by the
Bank","letterOfCreditNo":"LOC422344842786394989"},{"transactionId":"TX422345235524244330","tr
ansactionType":"ReleasePayment","timeStamp":"2022-08-23 15:02:57.368495005 +0000 UTC
m=+27369.628729618","transactionInitiatedBy":"Bank","remarks":"Amount paid
successfully","letterOfCreditNo":"LOC422344842786394989"},

```

The screenshot shows the Eclipse IDE's console window. At the top, there are two tabs for 'ClientApp.java'. Below the tabs, the console output is visible. It starts with a red line indicating a terminated Java application. This is followed by three log4j warnings in red text: 'No appenders could be found for logger (org.hyperledger.fabric.sdk.helper.Config)', 'Please initialize the log4j system properly.', and a link to the log4j FAQ. Then, the text 'ReleasePayment: Action Failed' appears. Finally, a JSON object is printed, representing a failed transaction.

```
<terminated> ClientApp [Java Application] C:\Users\rachana.sundar\Downloads\eclipse-java-2022-06-R-win32-x86_64\eclipse\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64.jre\bin\java.exe
log4j:WARN No appenders could be found for logger (org.hyperledger.fabric.sdk.helper.Config).
log4j:WARN Please initialize the log4j system properly.
log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more info.
ReleasePayment: Action Failed
{"buyer":"hcl","dateOfCreation":"2022-08-23 14:59:03.297825256 +0000 UTC m=+27134.513812367","buyerBank":"hdfc","seller":"lene
```

```
{"transactionId":"TX422345428260901738","transactionType":"ReleasePayment","timeStamp":"2022-08-23 15:04:52.246756096 +0000 UTC m=+27484.506944955","transactionInitiatedBy":"Bank","remarks":"Failed to pay amount","letterOfCreditNo":"LOC422344842786394989"}}}
```