KYC Case Study Functionality Testing Document

Assignment : KYC Case Study

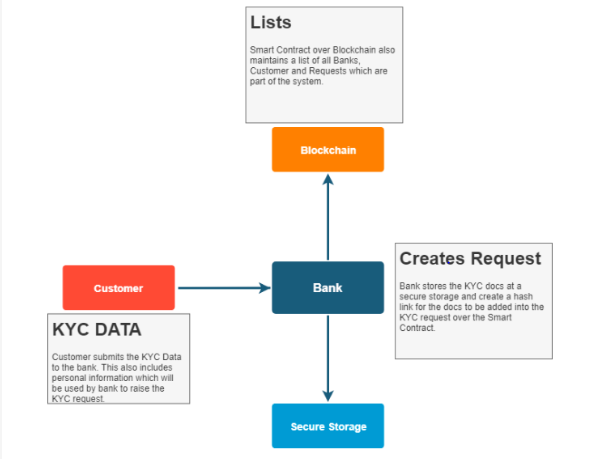
Submission to : Upgrade

submitted by : Archana

1. KYC Use Case

The source for the specifications of detailed KYC Case study is provided in the assignment module of course 6(Fundamentals of BlockChain and Smart Contracts).

KYC Use Case Block Diagram



Testing Environment : Remix Ide in an online Environment : Plugins - solidity complier, Deploy and Run Transactions

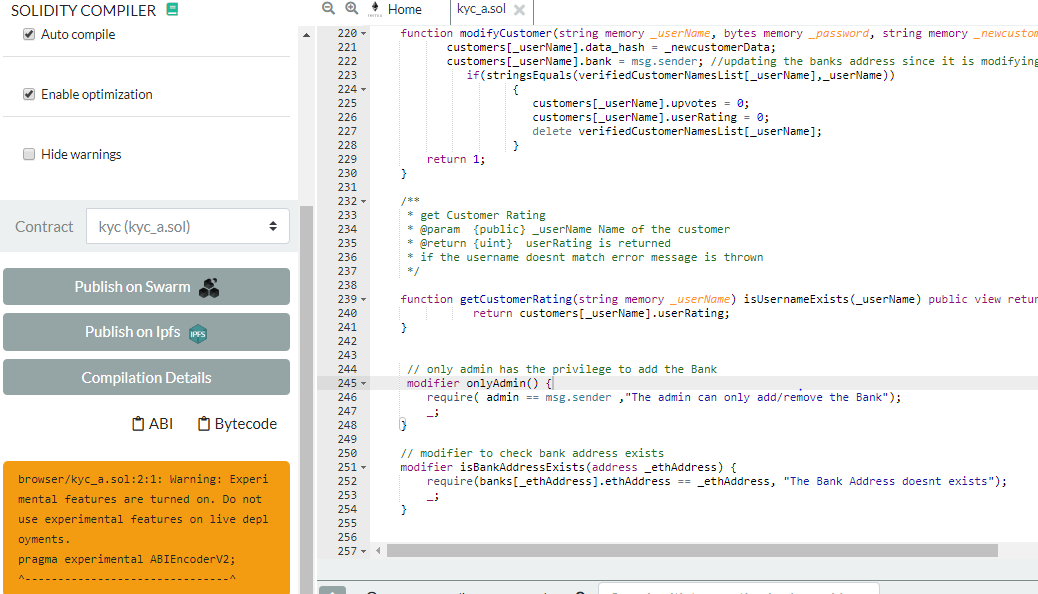
|  |  |
| --- | --- |
|  |  |

1. List of functions

|  |  |
| --- | --- |
| S.No | Name of the function |
| 1 | Add Bank |
| 2 | Remove Bank |
| 3 | upvoteBank |
| 4 | getBankRating |
| 5 | getBankDetails |
| 6 | BankCount |
| 7 | addKYCRequest |
| 8 | removeKYCRequest |
| 9 | getBankInitiatedKYCRequest |
| 10 | addCustomer |
| 11 | removeCustomer |
| 12 | setPassword |
| 13 | viewCustomer |
| 14 | upVoteCustomer |
| 15 | getCustomerRating |
| 16 | modifycustomer |
| 17 | getAccessHistory |

1. compile the smart contract

The smart contract compiled successfully with one warning because of using abi experimental



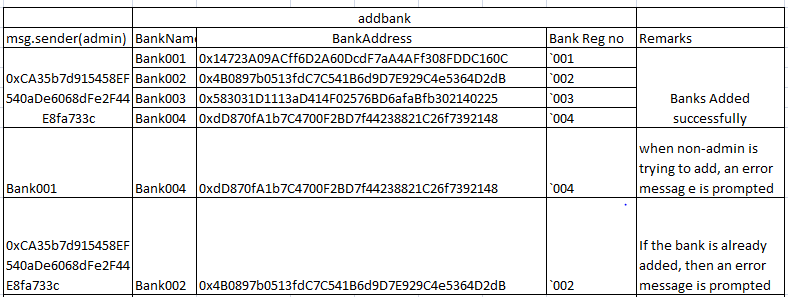
2. Deploy the smart contract

Admin Adress: 0xCA35b7d915458EF540aDe6068dFe2F44E8fa733c

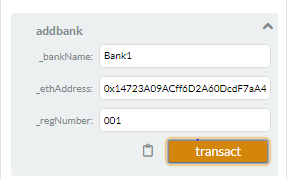


3. addBank

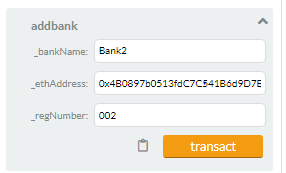
|  |  |
| --- | --- |
| Functionality | Remarks |
| This function is used by the admin to add a bank to the KYC Contract. You need to verify if the user trying to call this function is admin or not. | 1.Added 4 banks successfully  2.when non-admin is trying to add, an error messag e is prompted  3.If the bank is already added, then an error message is prompted |



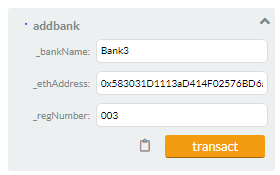
Adding four Banks:



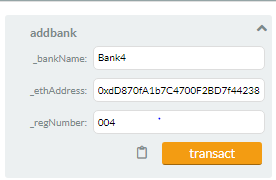






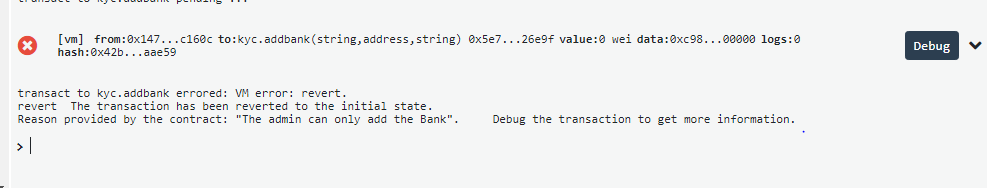




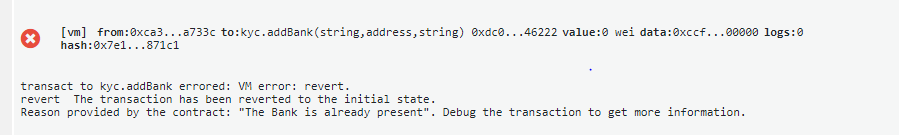




Bank1 (non -admin) is trying to add Bank4 an error message is prompted

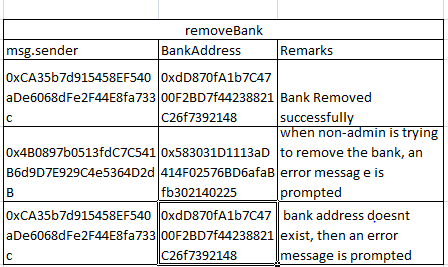


Bank2 is already present, when trying to add the same Bank an error message is prompted

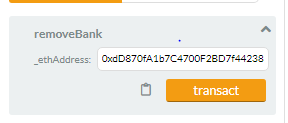


4. removeBank

|  |  |
| --- | --- |
| Functionality | Remarks |
| This function is used by the admin to remove a bank from the KYC Contract.  You need to verify if the user trying to call this function is admin or not. | 1.removed bank successfully  2.when non-admin is trying to remove the bank, an error messag e is prompted  3.If the bank address doesnt exist, then an error message is prompted |

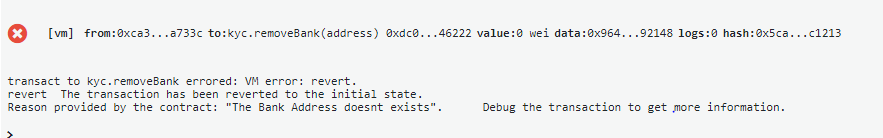


remove the Bank004 with address 0xdD870fA1b7C4700F2BD7f44238821C26f7392148



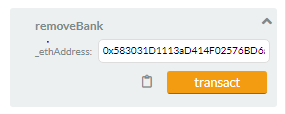


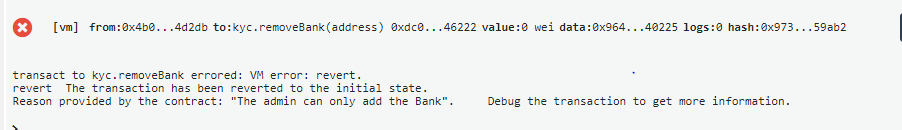
Trying to remove the Bank004 whose address is not found : Error message is prompted



Bank002(non admin) is trying to remove the Bank003 : Error message is prompted

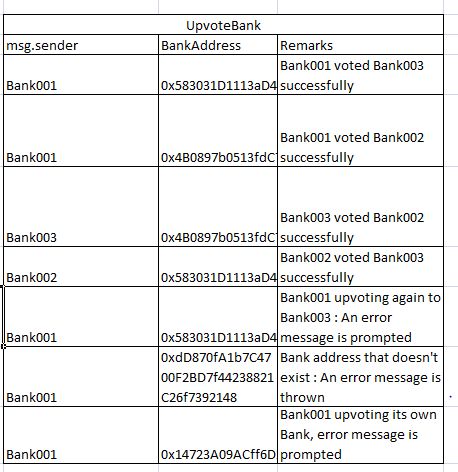




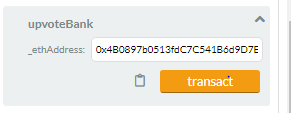


5. upvoteBank

|  |  |
| --- | --- |
| Functionality | Remarks |
| This function is used to add and update votes for the banks. You also need to update the rating for the bank in this function. If the bank has already voted for the same bank before, reject the request. | 1.Bank upvotes successfully  2Bank voted for the same bank, an error messag e is prompted  3.If the bank address doesnt exist, then an error message is prompted  4.A Bank cannot upvote its own bank, then an error message is prompted |

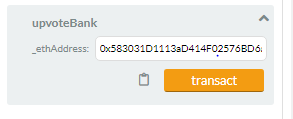


Bank001 upvoting Bank002



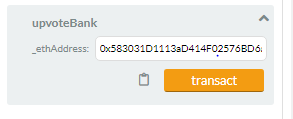


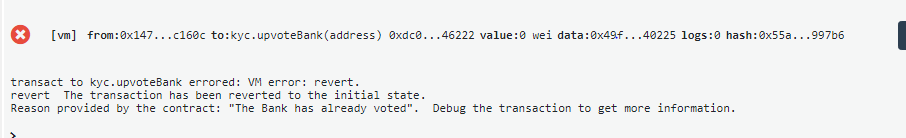
Bank001 upvoting Bank003



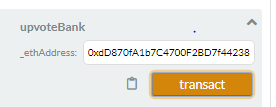


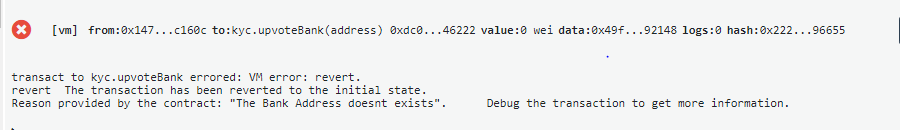
Bank001 upvoting again to Bank003 : An error message is prompted



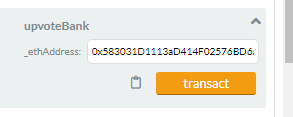


Checking for the Bank address that doesn't exist : An error message is thrown



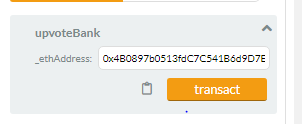


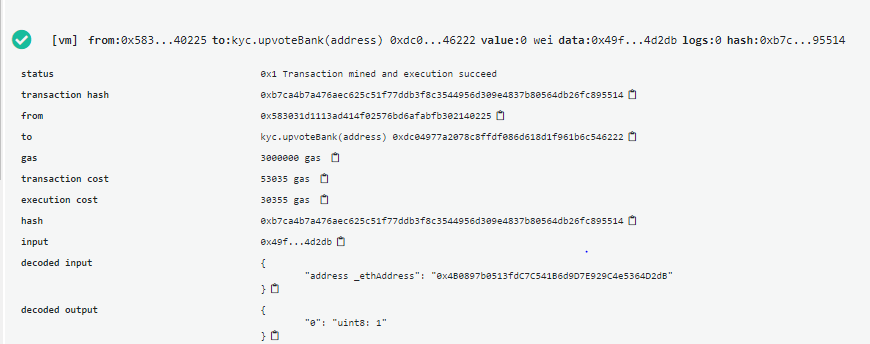
Bank002 upvoting for Bank003



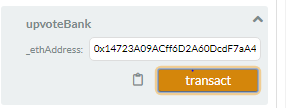


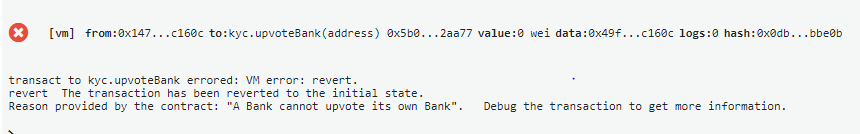
Bank3 upvoting for Bank002





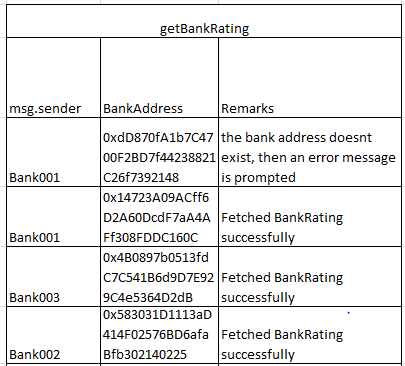
Bank001 upvoting its own Bank, error message is prompted



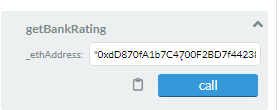


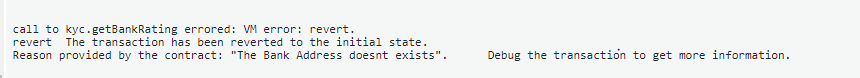
6. getBankRating

|  |  |
| --- | --- |
| Functionality | Remarks |
| This function is used to fetch bank rating from the smart contract | 1.fetched bank rating successfully  2. If the bank address doesnt exist, then an error message is prompted |



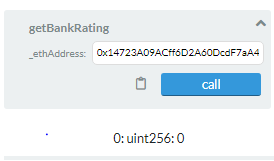
checking for the BankAddress that doesnt exists





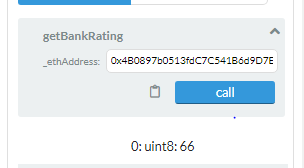
BankRating for Bank 001:

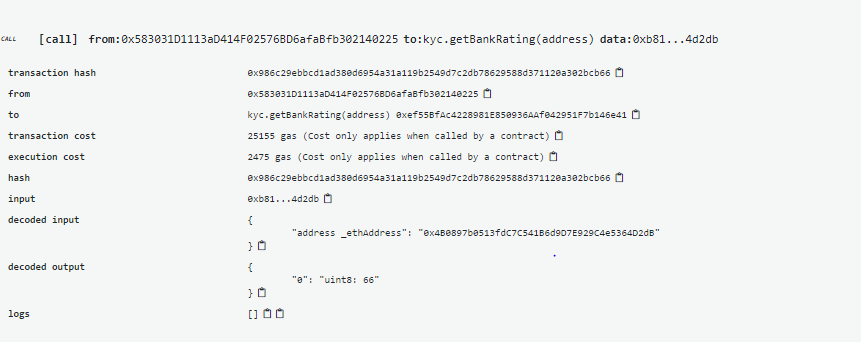
Note: No bank has voted for Bank001



BankRating for Bank 002:

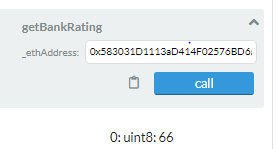
Note : Bank001 and Bank003 has voted

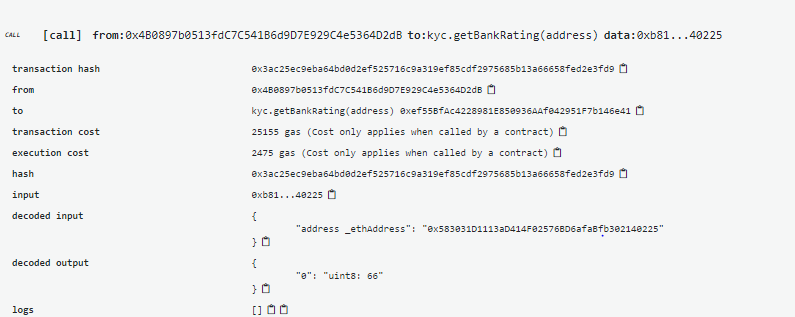




BankRating for Bank 003:

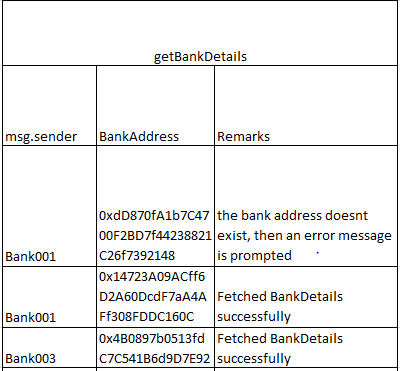
Note : Bank001 and Bank002 has voted



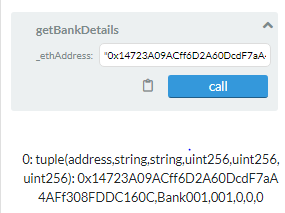


7. getBankDetails

|  |  |
| --- | --- |
| Functionality | Remarks |
| This function is used to fetch the bank details. | 1.fetched bank details successfully  2. If the bank address doesnt exist, then an error message is prompted |

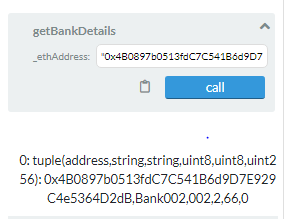


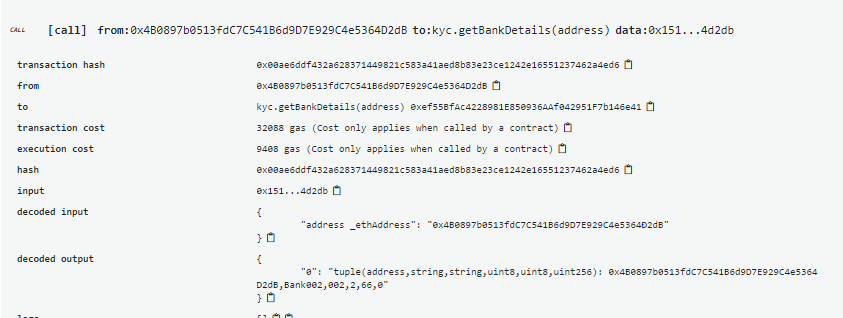
Bank001 Details:



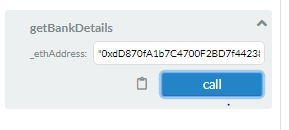


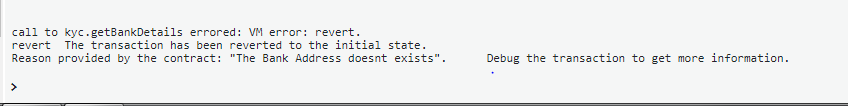
Bank002 Details:



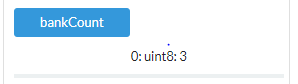


checking for the Bankaddress that doesnt exist: An error message is thrown



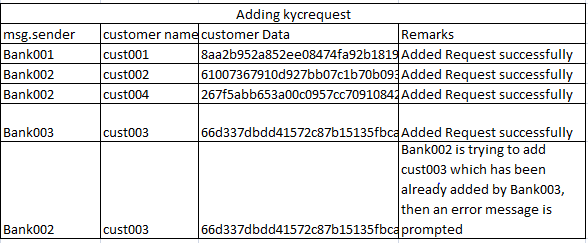


8. bankCount : The no. of bankcount variable is also updating correctly



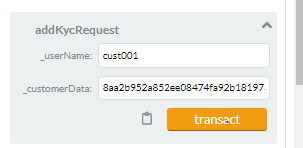
9. AddKYCRequest

|  |  |
| --- | --- |
| Functionality | Remarks |
| This function is used to add the KYC request to the requests list. If the bank rating is less than or equal to 0.5 then assign IsAllowed to false. Else assign IsAllowed to true. . | 1.added kyc requests successfully  2.The customer data already exists and if other bank is trying to process it, then an error message is prompted(This is to avoid the duplication process) |



Adding cust001, 8aa2b952a852ee08474fa92b181973f397b91d8dee467ee84db9bc5a82c62bee

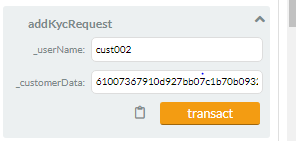
This cust001 is been added by this Bank001: 0x14723A09ACff6D2A60DcdF7aA4AFf308FDDC160C





Adding cust002, 61007367910d927bb07c1b70b093265d4d7f44c1672b21d12a560f4392783d5c

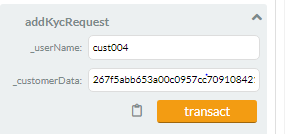
This cust002 is been added by this Bank002:0x4B0897b0513fdC7C541B6d9D7E929C4e5364D2dB





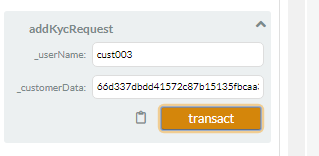
Adding cust004, 267f5abb653a00c0957cc70910842174a877fcfcc4a781ae99ecaaa1a87b039b

This cust004 is been added by this Bank002:0x4B0897b0513fdC7C541B6d9D7E929C4e5364D2dB



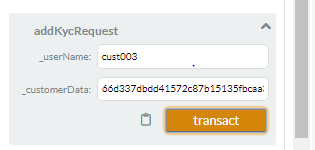
Adding cust003, 66d337dbdd41572c87b15135fbcaa3b5e36b9949409cf7104f2fe7ccdd035dcb

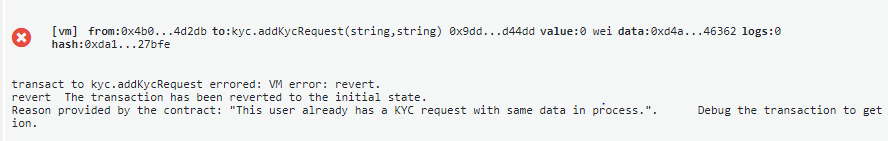
This cust003 is been added by this Bank003 : 0x583031D1113aD414F02576BD6afaBfb302140225





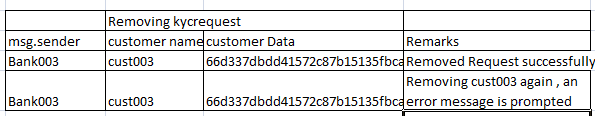
Bank002 is trying to add cust003 which has been already added by Bank003, then an error message is prompted



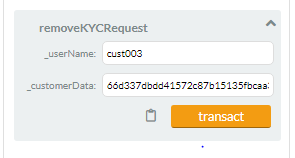


10. RemoveKYCRequest

|  |  |
| --- | --- |
| Functionality | Remarks |
| This function will remove the request from the requests list. | 1.removed kyc request successfully  2.If the customer data doesnt exist, then an error message is prompted |

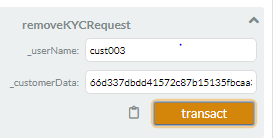


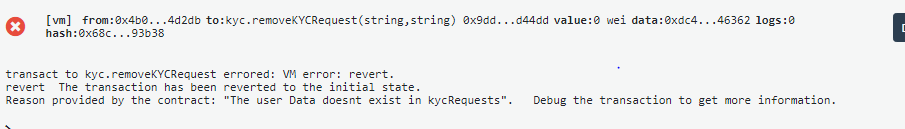
Removing cust003





Removing cust003 again , an error message is prompted

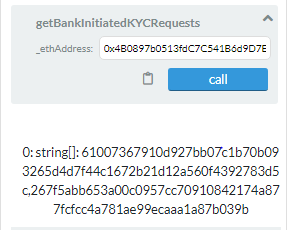




11. getBankInitiatedKYCRequest

|  |  |
| --- | --- |
| Functionality | Remarks |
| This function fetches the KYC requests for a specific bank | 1.fectched the list of kyc requests initiated by bank 002 successfully |

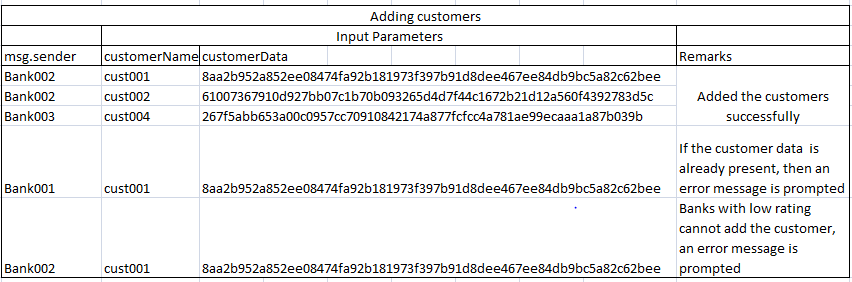
List of requests raised by Bank002



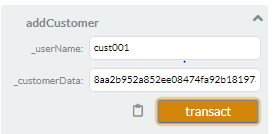


12. addCustomer

|  |  |
| --- | --- |
| Functionality | Remarks |
| This function will add a customer to the customer list. If IsAllowed is false then don't process the request. | 1.added the customers successfully  2.If the customer data is already present, then an error message is prompted  3.Banks with low rating cannot add the customer, an error message is prompted |

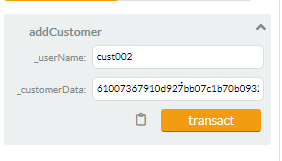


Adding cust001 by Bank002



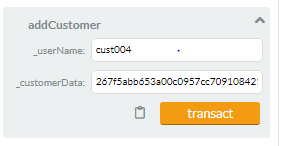


Adding cust002 by Bank002



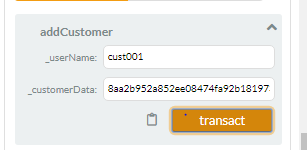


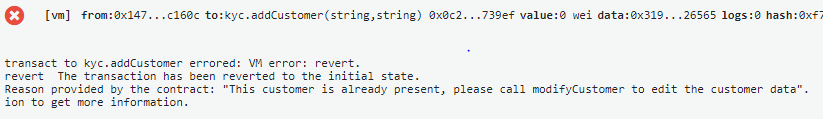
Adding cust004 by Bank003



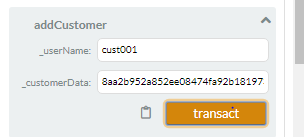


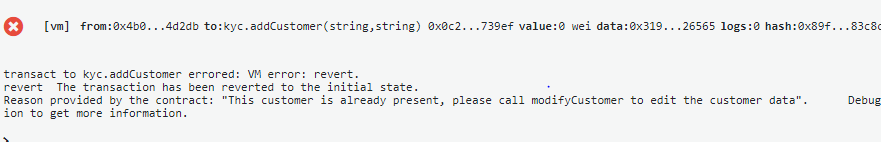
Bank001 is trying to add the cust001, an error message is prompted because of low rating





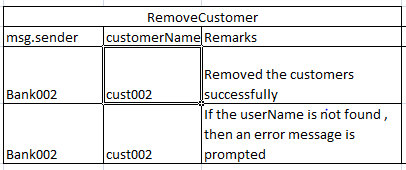
Bank002 is trying to add already added cust001,an error message is prompted



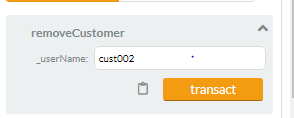


13. removeCustomer

|  |  |
| --- | --- |
| Functionality | Remarks |
| This function will remove the customer from the customer list | 1.removed the customer successfully  2.If the userName is not found , then an error message is prompted |

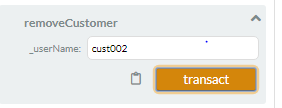


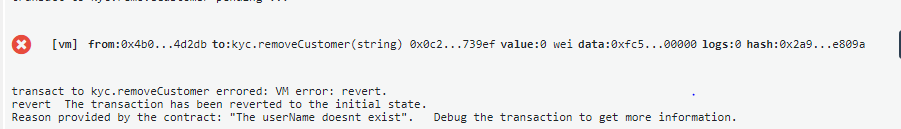
Removed the cust002 successfully





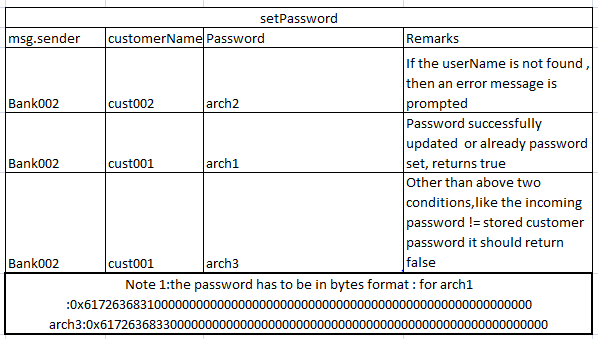
Deleting the cust002 again, an error message is prompted



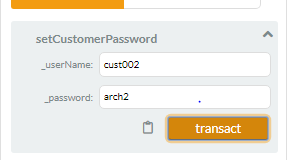


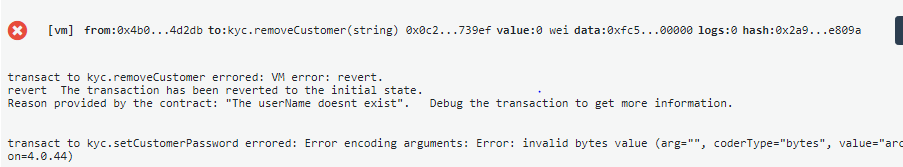
14. setPassword

|  |  |
| --- | --- |
| Functionality | Remarks |
| This function is used to set a password for customer data, which can be later be unlocked by using the password. This required to give access to only certain banks. It could be a possibility that customers do not want to share his/her data with everyone. To make sure the customer data is protected, we add a password to customer data. While retrieving the information for customer, bank also needs to pass the password, if the password is set for the customer. If the password is not set then every bank can view the data. | 1. .If the userName is not found , then an error message is prompted  2. updated password successfully  3. If the password is already set, then it returns false |

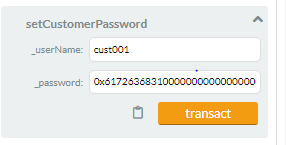


The userName is not found , an error message is prompted



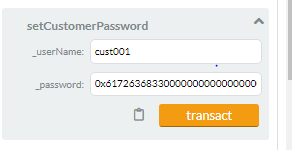


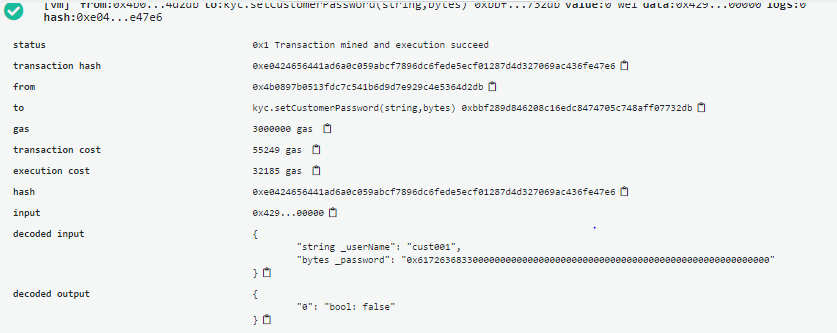
The password wasnt set, updated successfully or if the password already set ---returns true





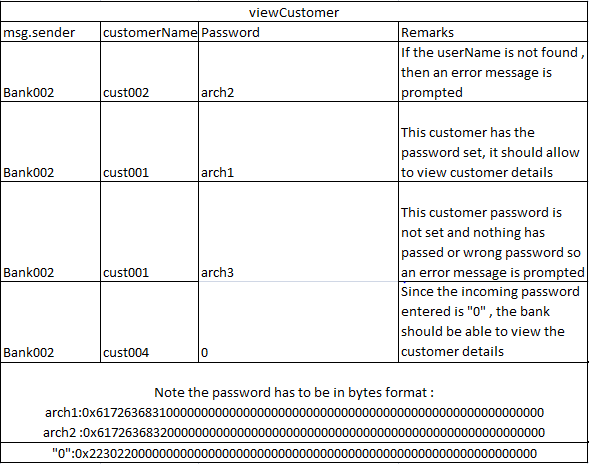
Since password for cust001 is wrongly entered, it is retuning false



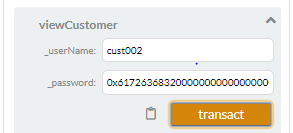


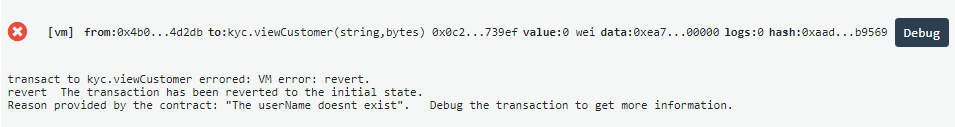
15. viewCustomer

|  |  |
| --- | --- |
| Functionality | Remarks |
| This function allows a bank to view details of a customer | 1. If the userName is not found , then an error message is prompted  2. The customer has the password set, it should allow to view customer details  3. This customer password is not set and nothing has passed or an wrong password so an error message is prompted  4. Since the incoming password entered is "0" , the bank should be able to view the customer details |

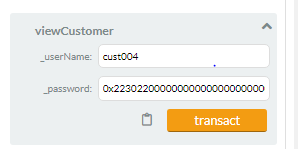


The userName is not found , an error message is prompted

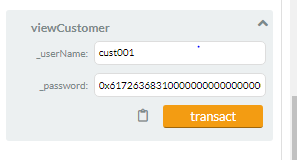




Since the incoming password entered is "0" , the bank should be able to view the customer details

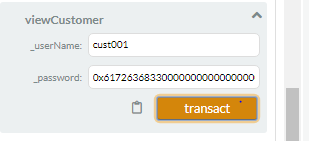
 

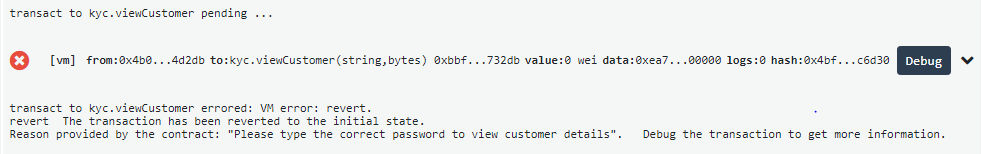
This customer has the password set, it should allow to view customer details





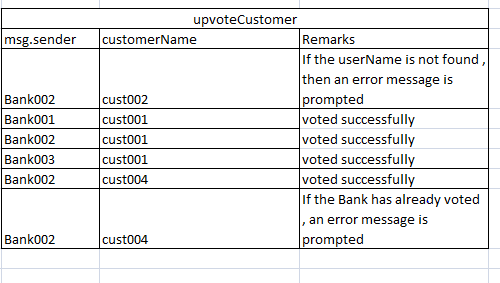
This customer password is not set and nothing has passed or a wrong password so an error message is prompted



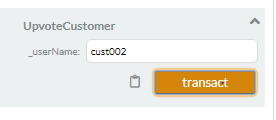


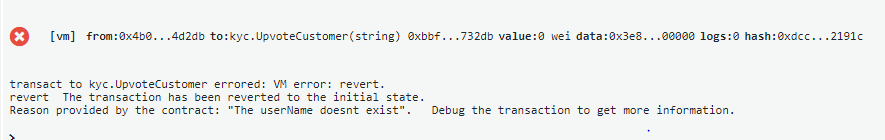
16. upvoteCustomer

|  |  |
| --- | --- |
| Functionality | Remarks |
| This function allows a bank to cast an upvote for a customer. This vote from a bank means that it accepts the customer details as well acknowledge the KYC process done by some bank on the customer. You also need to update the rating for a customer in this function. The rating is calculated as the number of upvotes for the customer/total number of banks. If rating is more than 0.5, then you can add the customer to the final\_customer list. If the bank has already voted for the same customer before, reject the request. | 1. If the userName is not found , then an error message is prompted  2.If the Bank has already voted , an error message is prompted  3.banks upvoted the custmers successfully |

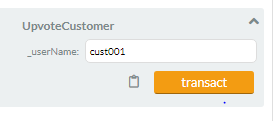


If the userName is not found , then an error message is prompted



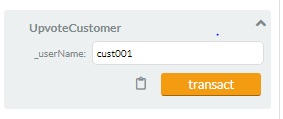


Bank002 voting for cust001:



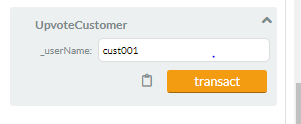


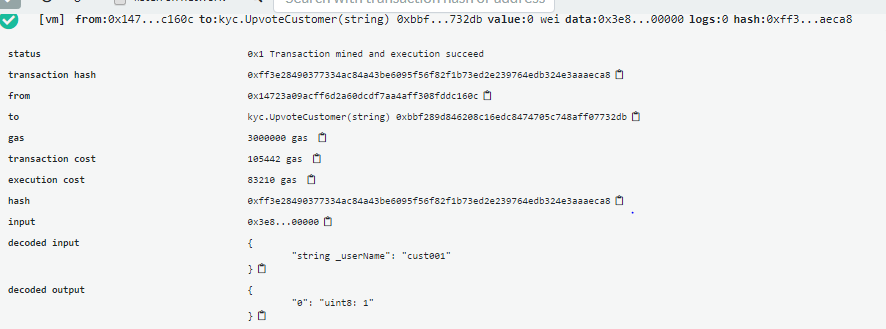
Bank001 voting for cust001:



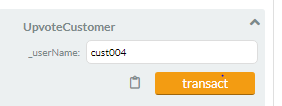


Bank003 voting for cust001:



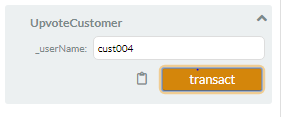


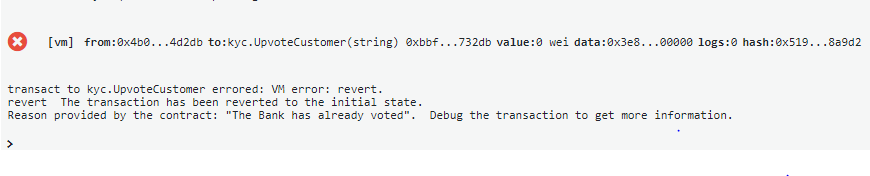
Bank002 voting for cust004:





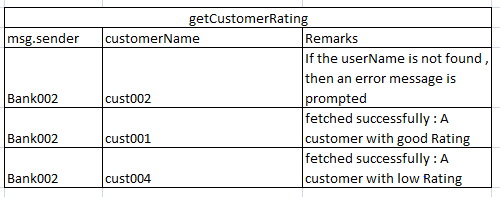
If the Bank004 trying to vote again cust004 , an error message is prompted



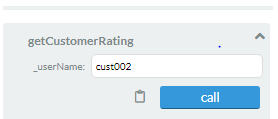


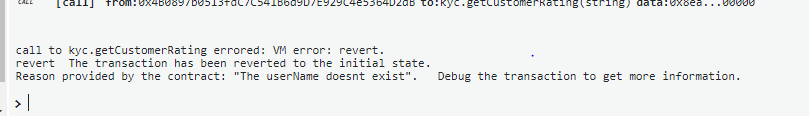
17. getCustomerRating

|  |  |
| --- | --- |
| Functionality | Remarks |
| This function is used to fetch customer rating from the smart contract. | 1. If the userName is not found , then an error message is prompted  2.Fetched the bank rating successfully |

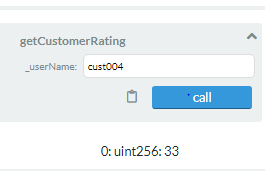


If the userName is not found , then an error message is prompted





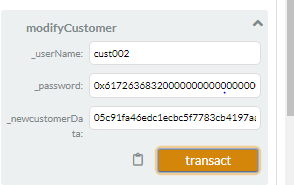
cust004 BankRating

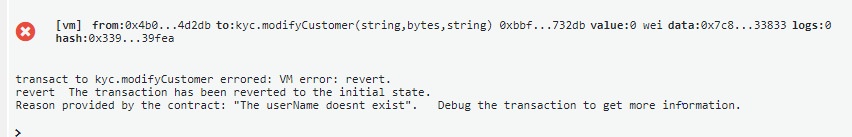


18. modifyCustomer

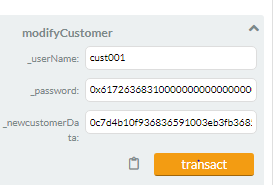
|  |  |
| --- | --- |
| Functionality | Remarks |
| This function allows a bank to modify a customer's data. Only applicable for the customers whose request have been validated and present in the customer list. If the user is present in the final customer list then remove it from the final list and change the upvotes and rating component of the customer in customer list to "0". Remove all the previous upvotes for the customer. Hence, banks need to again upvote on the customer to acknowledge the modified data. | 1. If the userName is not found , then an error message is prompted  2.modified the customer |

If the userName is not found , then an error message is prompted





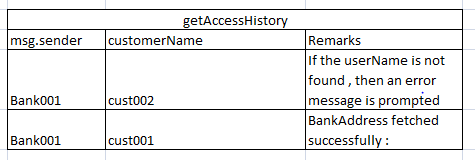
modified the customer: cust001



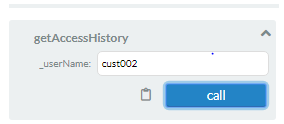


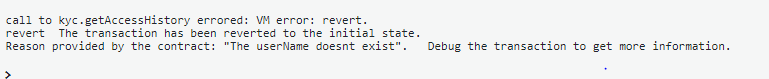
18. getAccessHistory

|  |  |
| --- | --- |
| Functionality | Remarks |
| This function is used to fetch the bank details which made the last changes to the customer data. | 1. If the userName is not found , then an error message is prompted  2.on successful retrieval bank address is returned |

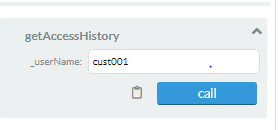


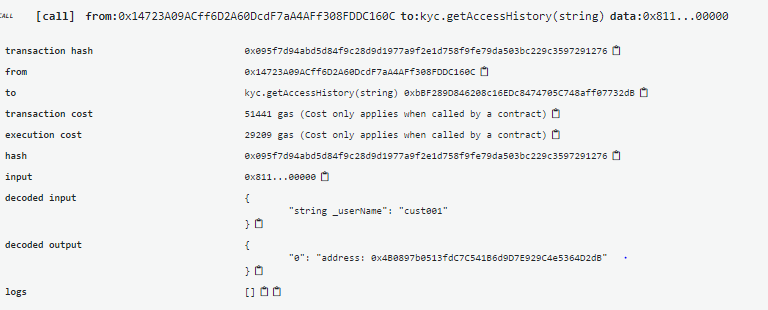
If the userName is not found , then an error message is prompted





cust001 bankaddress fetched successfully.





Result : The functionality over multiple test checks is been done, and achieved as per the assignment expected results

References: Upgrade course content, stackover exchage, udemy Tomweisner lectures, solidity documentation, https://www.bitdegree.org/learn/solidity-events etc

Future Enhancements :----to do-------