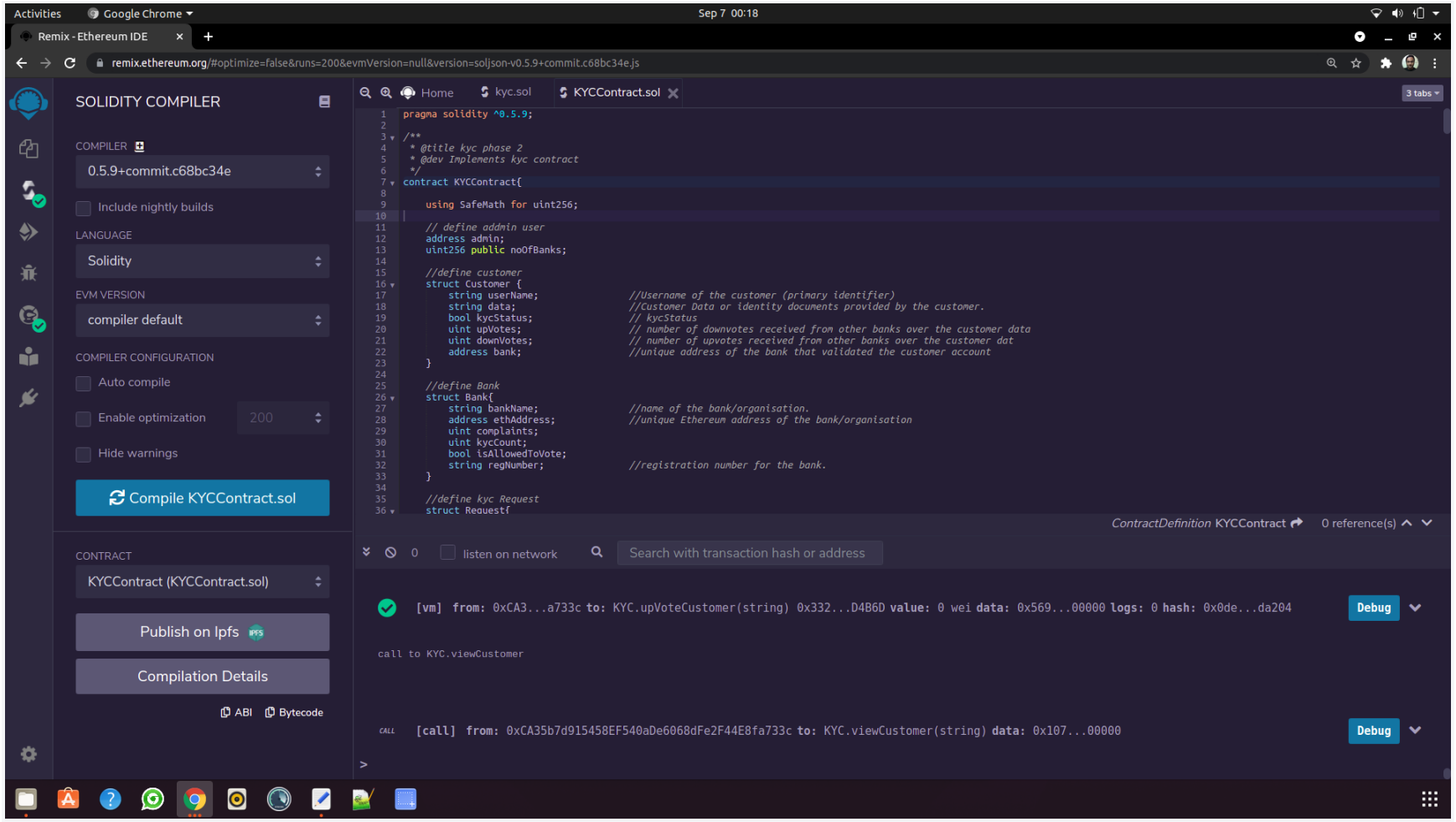
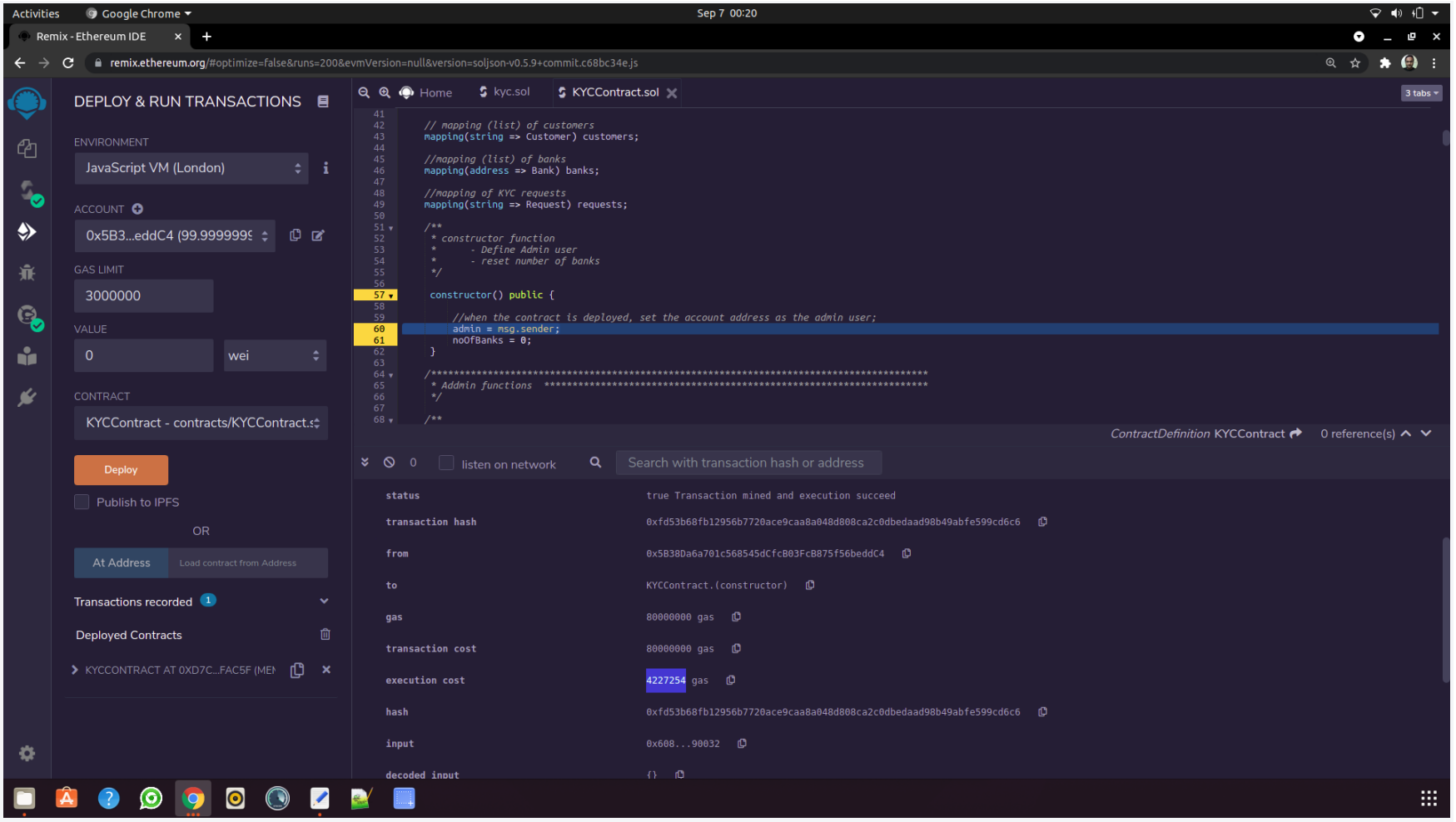
Instructions KYCContract.sol:

Testing KYCcontract functions on Remix:

1) Contract compiled successfully:



2) Deploy Contract on Java script VM:



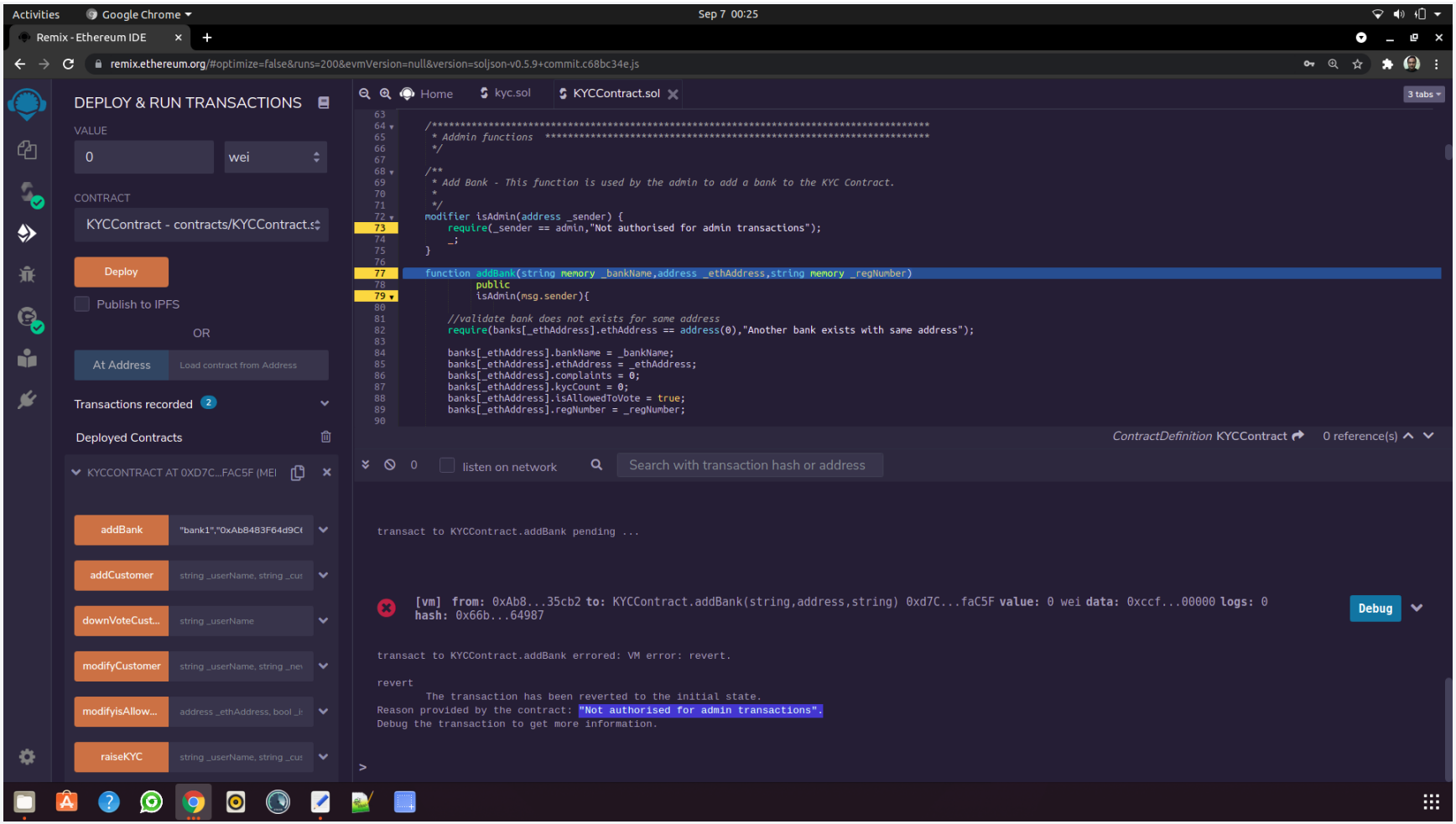
Used test address provided by the remix environment for testing:

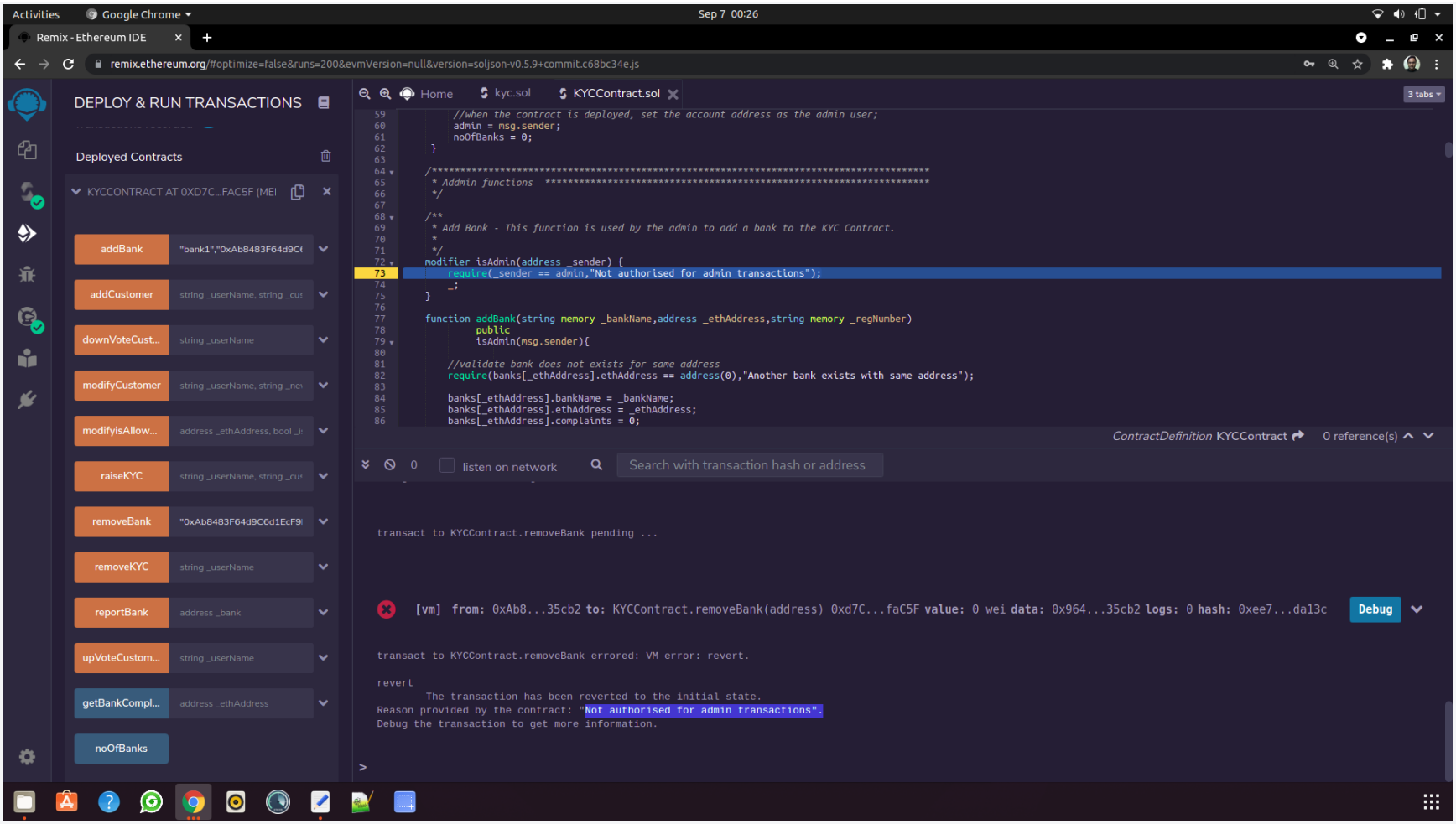
admin: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4

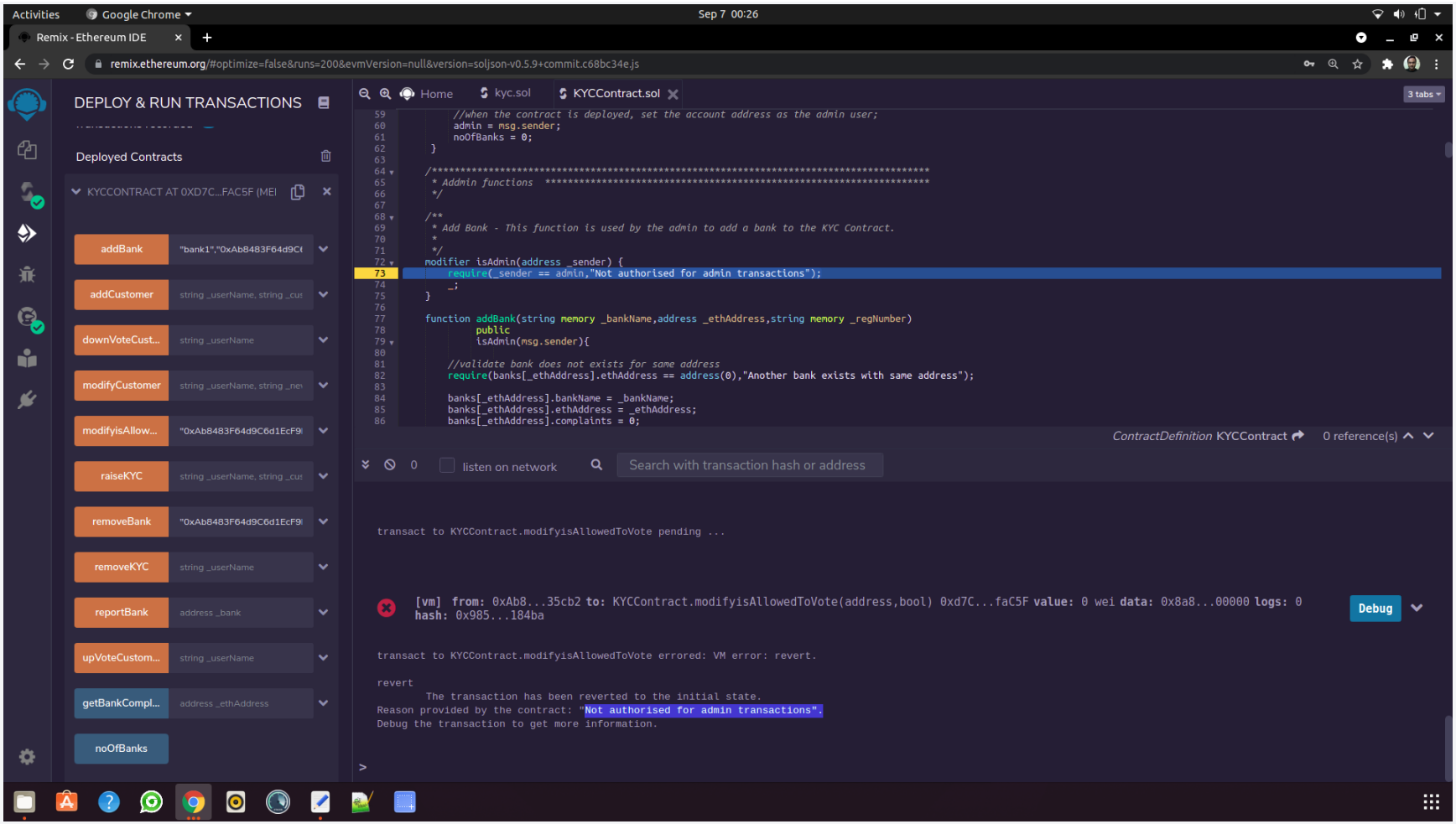
Bannk Name: Address - regNo.
bank1: 0xAb8483F64d9C6d1EcF9b849Ae677dD3315835cb2 - reg1
bank2: 0xCA35b7d915458EF540aDe6068dFe2F44E8fa733c - reg2
bank3: 0x4B20993Bc481177ec7E8f571ceCaE8A9e22C02db - reg3
bank4: 0x78731D3Ca6b7E34aC0F824c42a7cC18A495cabaB - reg4
bank5: 0x617F2E2fD72FD9D5503197092aC168c91465E7f2 - reg5
bank6: 0x17F6AD8Ef982297579C203069C1DbfFE4348c372 - reg6

1) Admin Functionality

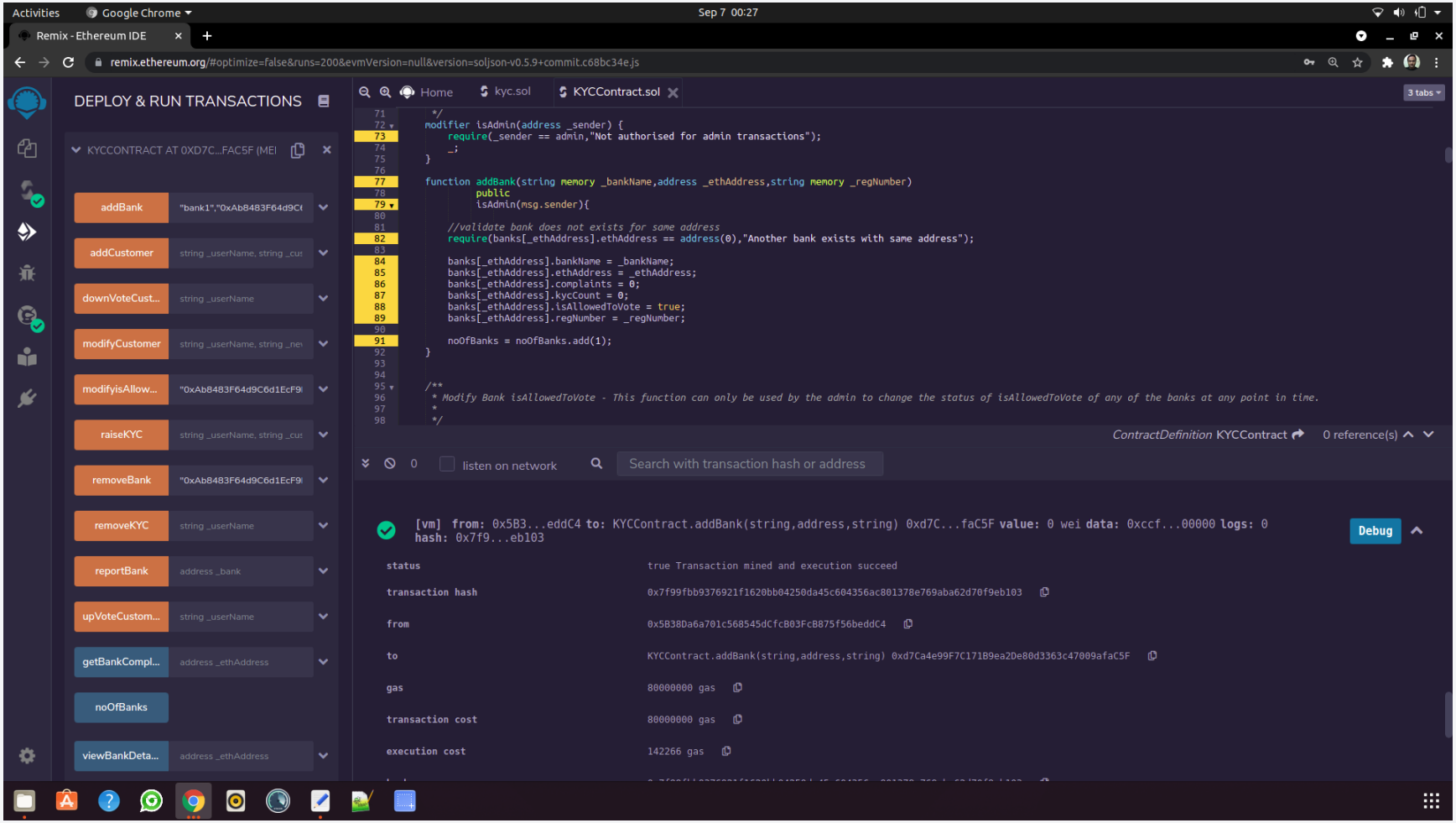
-- only admin can add Bank or remove bank or update voting rights


Activities
Google Chrome
Sep 7 00:25
Remix - Ethereum IDE
remix.ethereum.org/#optimize=false&runs=200&evmVersion=null&version=soljson-v0.5.9+commit.c68bc34e.js
DEPLOY & RUN TRANSACTIONS
VALUE
0
wei
CONTRACT
KYCContract - contracts/KYCContract.s
Deploy
Publish to IPFS
OR
At Address
Load contract from Address
Transactions recorded 2
Deployed Contracts
KYCCONTRACT AT 0XD7C...FAC5F (MEI
addBank
"bank1","0xAb8483F64d9C
addCustomer
string _userName, string _cus
downVoteCust...
string _userName
modifyCustomer
string _userName, string _ne
modifyisAllow...
address _ethAddress, bool _i
raiseKYC
string _userName, string _cus
Home
kyc.sol
KYCContract.sol
3 tabs
63
64 /*****
65 * Addmin functions *****
66 */
67
68 /**
69 * Add Bank - This function is used by the admin to add a bank to the KYC Contract.
70 *
71 */
72 modifier isAdmin(address _sender) {
73 require(_sender == admin,"Not authorised for admin transactions");
74 _;
75 }
76
77 function addBank(string memory _bankName,address _ethAddress,string memory _regNumber)
78 public
79 isAdmin(msg.sender){
80
81 //validate bank does not exists for same address
82 require(banks[_ethAddress].ethAddress == address(0),"Another bank exists with same address");
83
84 banks[_ethAddress].bankName = _bankName;
85 banks[_ethAddress].ethAddress = _ethAddress;
86 banks[_ethAddress].complaints = 0;
87 banks[_ethAddress].kycCount = 0;
88 banks[_ethAddress].isAllowedToVote = true;
89 banks[_ethAddress].regNumber = _regNumber;
90
ContractDefinition KYCContract
0 reference(s)
0
listen on network
Search with transaction hash or address
transact to KYCContract.addBank pending ...
[vm] from: 0xAb8...35cb2 to: KYCContract.addBank(string,address,string) 0xd7C...faC5F value: 0 wei data: 0xccf...00000 logs: 0
hash: 0x66b...64987
Debug
transact to KYCContract.addBank errored: VM error: revert.
revert
The transaction has been reverted to the initial state.
Reason provided by the contract: "Not authorised for admin transactions".
Debug the transaction to get more information.
>

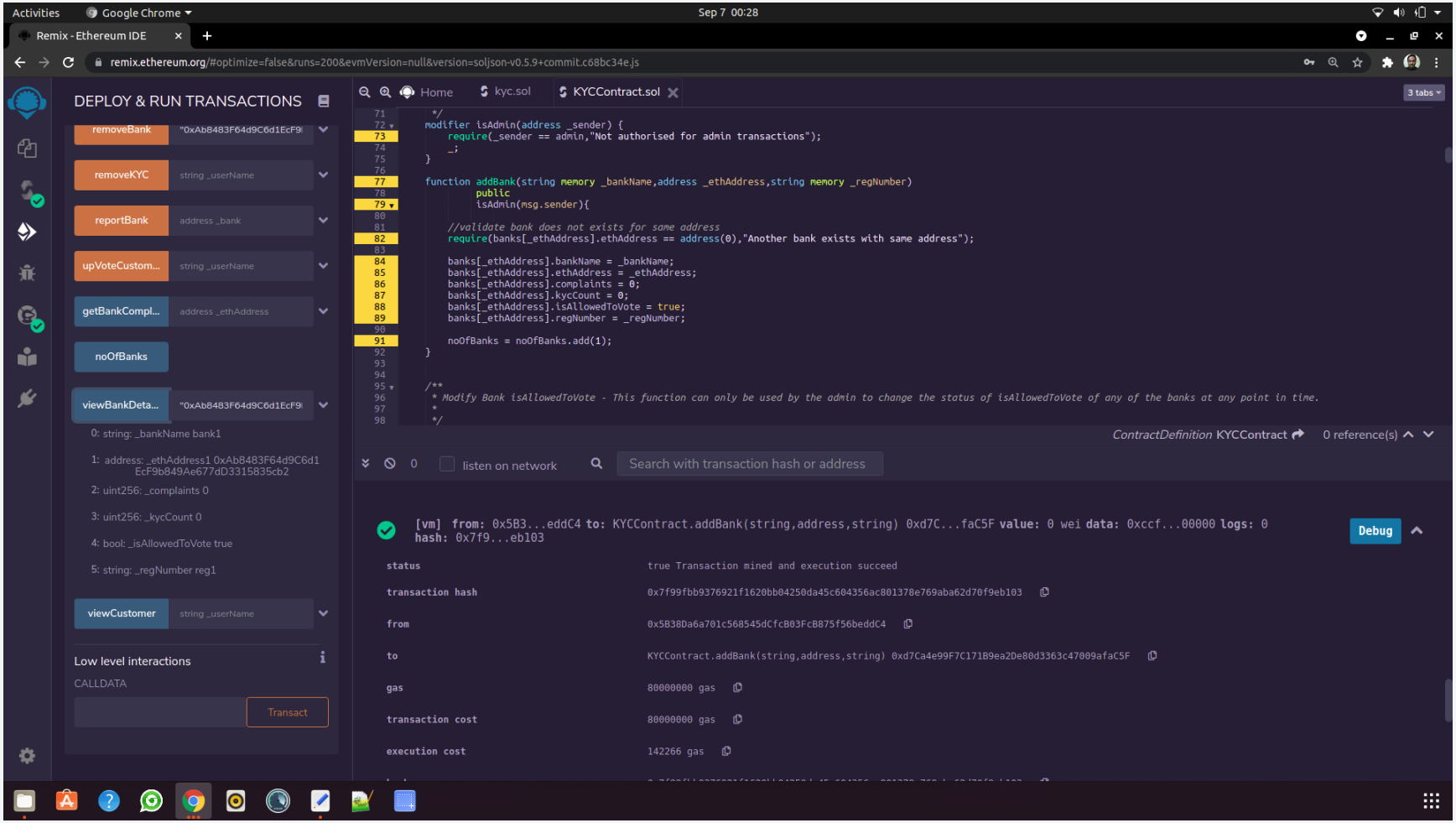

Activities
Google Chrome
Sep 7 00:26
Remix - Ethereum IDE
remix.ethereum.org/#optimize=false&runs=200&evmVersion=null&version=soljson-v0.5.9+commit.c68bc34e.js
DEPLOY & RUN TRANSACTIONS
Deployed Contracts
KYCCONTRACT AT 0XD7C...FAC5F (MEI
addBank
"bank1","0xAb8483F64d9C(
addCustomer
string _userName, string _cus
downVoteCust...
string _userName
modifyCustomer
string _userName, string _ne
modifyisAllow...
address _ethAddress, bool _i
raiseKYC
string _userName, string _cus
removeBank
"0xAb8483F64d9C6d1EcF9
removeKYC
string _userName
reportBank
address _bank
upVoteCustom...
string _userName
getBankCompl...
address _ethAddress
noOfBanks
Home
kyc.sol
KYCContract.sol
3 tabs
59 //when the contract is deployed, set the account address as the admin user;
60 admin = msg.sender;
61 noOfBanks = 0;
62 }
63
64 /*****
65 * Addmin functions *****
66 */
67
68 /**
69 * Add Bank - This function is used by the admin to add a bank to the KYC Contract.
70 *
71 */
72 modifier isAdmin(address _sender) {
73 require(_sender == admin,"Not authorised for admin transactions");
74 _;
75 }
76
77 function addBank(string memory _bankName,address _ethAddress,string memory _regNumber)
78 public
79 isAdmin(msg.sender){
80
81 //validate bank does not exists for same address
82 require(banks[_ethAddress].ethAddress == address(0),"Another bank exists with same address");
83
84 banks[_ethAddress].bankName = _bankName;
85 banks[_ethAddress].ethAddress = _ethAddress;
86 banks[_ethAddress].complaints = 0;
ContractDefinition KYCContract
0 reference(s)
0
listen on network
Search with transaction hash or address
transact to KYCContract.removeBank pending ...
[vm] from: 0xAb8...35cb2 to: KYCContract.removeBank(address) 0xd7C...faC5F value: 0 wei data: 0x964...35cb2 logs: 0 hash: 0xee7...da13c
Debug
transact to KYCContract.removeBank errored: VM error: revert.
revert
The transaction has been reverted to the initial state.
Reason provided by the contract: "Not authorised for admin transactions".
Debug the transaction to get more information.
>


Activities
Google Chrome
Sep 7 00:26
Remix - Ethereum IDE
remix.ethereum.org/#optimize=false&runs=200&evmVersion=null&version=soljson-v0.5.9+commit.c68bc34e.js
DEPLOY & RUN TRANSACTIONS
Deployed Contracts
KYCCONTRACT AT 0XD7C...FAC5F (MEI
addBank
"bank1","0xAb8483F64d9C
addCustomer
string _userName, string _cus
downVoteCust...
string _userName
modifyCustomer
string _userName, string _ne
modifyisAllow...
"0xAb8483F64d9C6d1EcF9
raiseKYC
string _userName, string _cus
removeBank
"0xAb8483F64d9C6d1EcF9
removeKYC
string _userName
reportBank
address _bank
upVoteCustom...
string _userName
getBankCompl...
address _ethAddress
noOfBanks
Home
kyc.sol
KYCContract.sol
3 tabs
59 //when the contract is deployed, set the account address as the admin user;
60 admin = msg.sender;
61 noOfBanks = 0;
62 }
63
64 /***
65 * Addmin functions ***
66 */
67
68 /**
69 * Add Bank - This function is used by the admin to add a bank to the KYC Contract.
70 *
71 */
72 modifier isAdmin(address _sender) {
73 require(_sender == admin,"Not authorised for admin transactions");
74 _;
75 }
76
77 function addBank(string memory _bankName,address _ethAddress,string memory _regNumber)
78 public
79 isAdmin(msg.sender){
80
81 //validate bank does not exists for same address
82 require(banks[_ethAddress].ethAddress == address(0),"Another bank exists with same address");
83
84 banks[_ethAddress].bankName = _bankName;
85 banks[_ethAddress].ethAddress = _ethAddress;
86 banks[_ethAddress].complaints = 0;
ContractDefinition KYCContract
0 reference(s)
0
listen on network
Search with transaction hash or address
transact to KYCContract.modifyisAllowedToVote pending ...
[vm] from: 0xAb8...35cb2 to: KYCContract.modifyisAllowedToVote(address,bool) 0xd7C...faC5F value: 0 wei data: 0x8a8...00000 logs: 0 hash: 0x985...184ba
Debug
transact to KYCContract.modifyisAllowedToVote errored: VM error: revert.
revert
The transaction has been reverted to the initial state.
Reason provided by the contract: "Not authorised for admin transactions".
Debug the transaction to get more information.
>

Admin is allowed to create new Bank



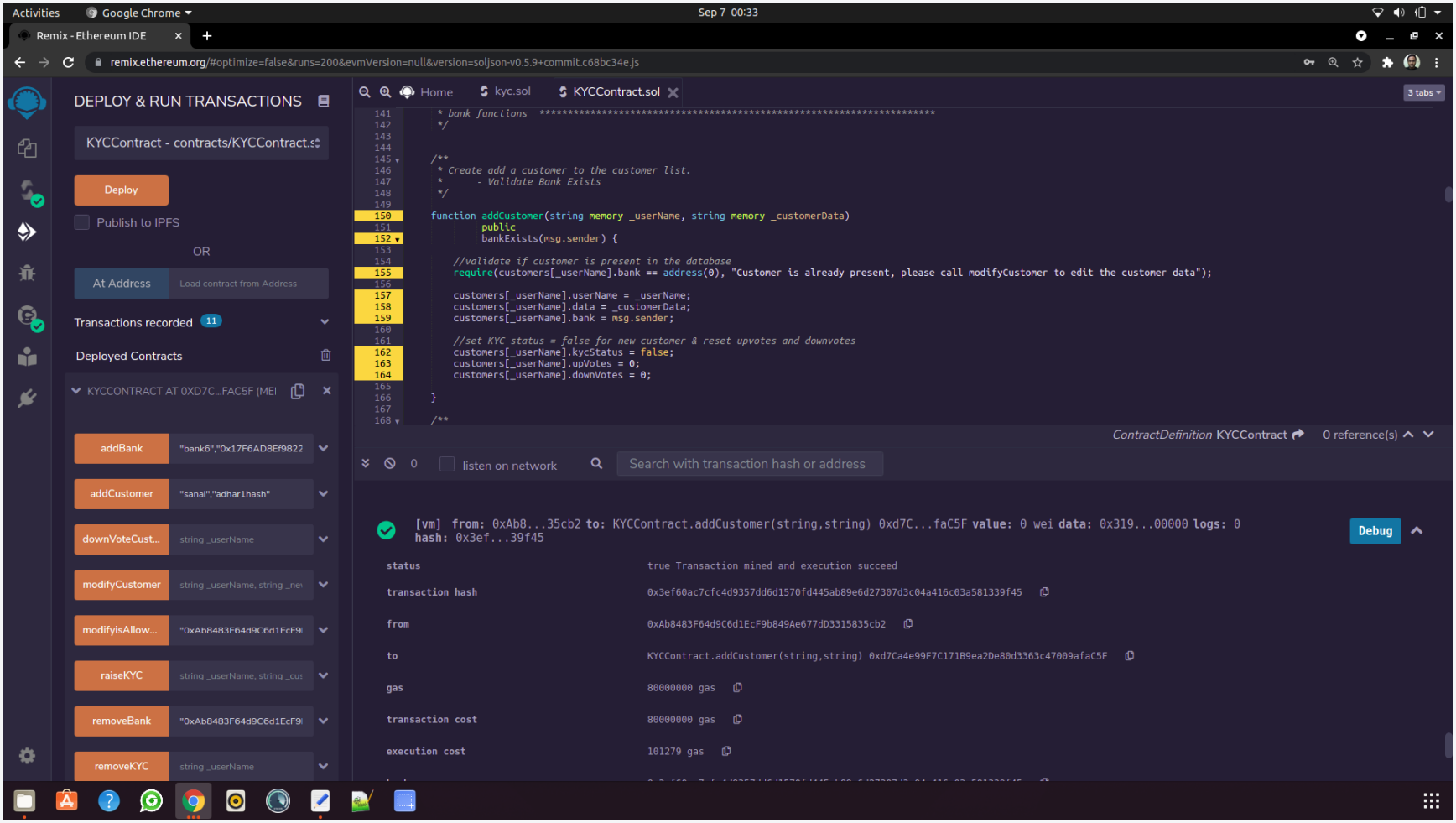
View Bank Details: (no of banks made public for testing only)



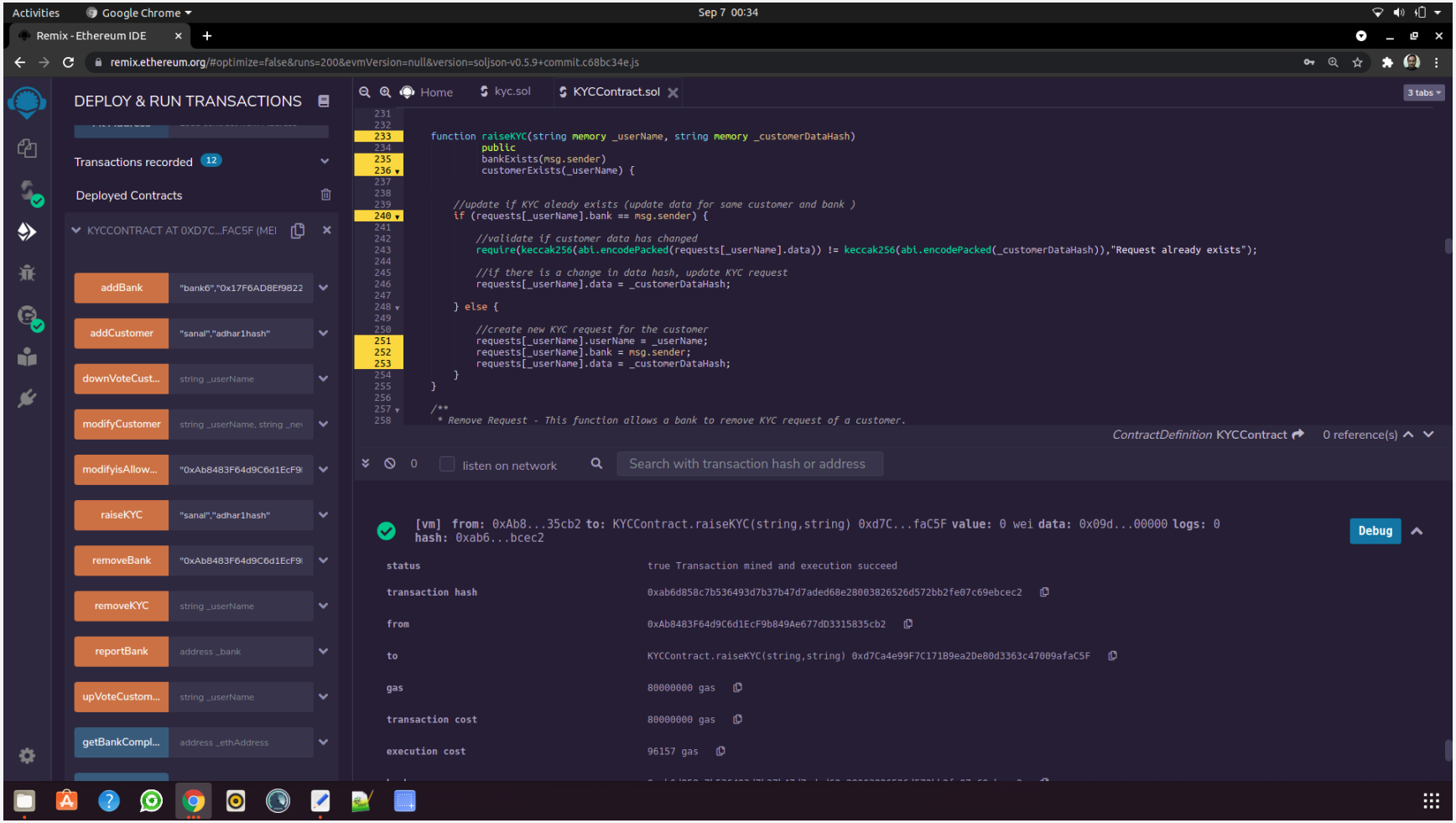
Added total 6 banks.

- noOfBanks counter made public for testing
- Logic for ⅓ rd banks only possible if number of banks > 5

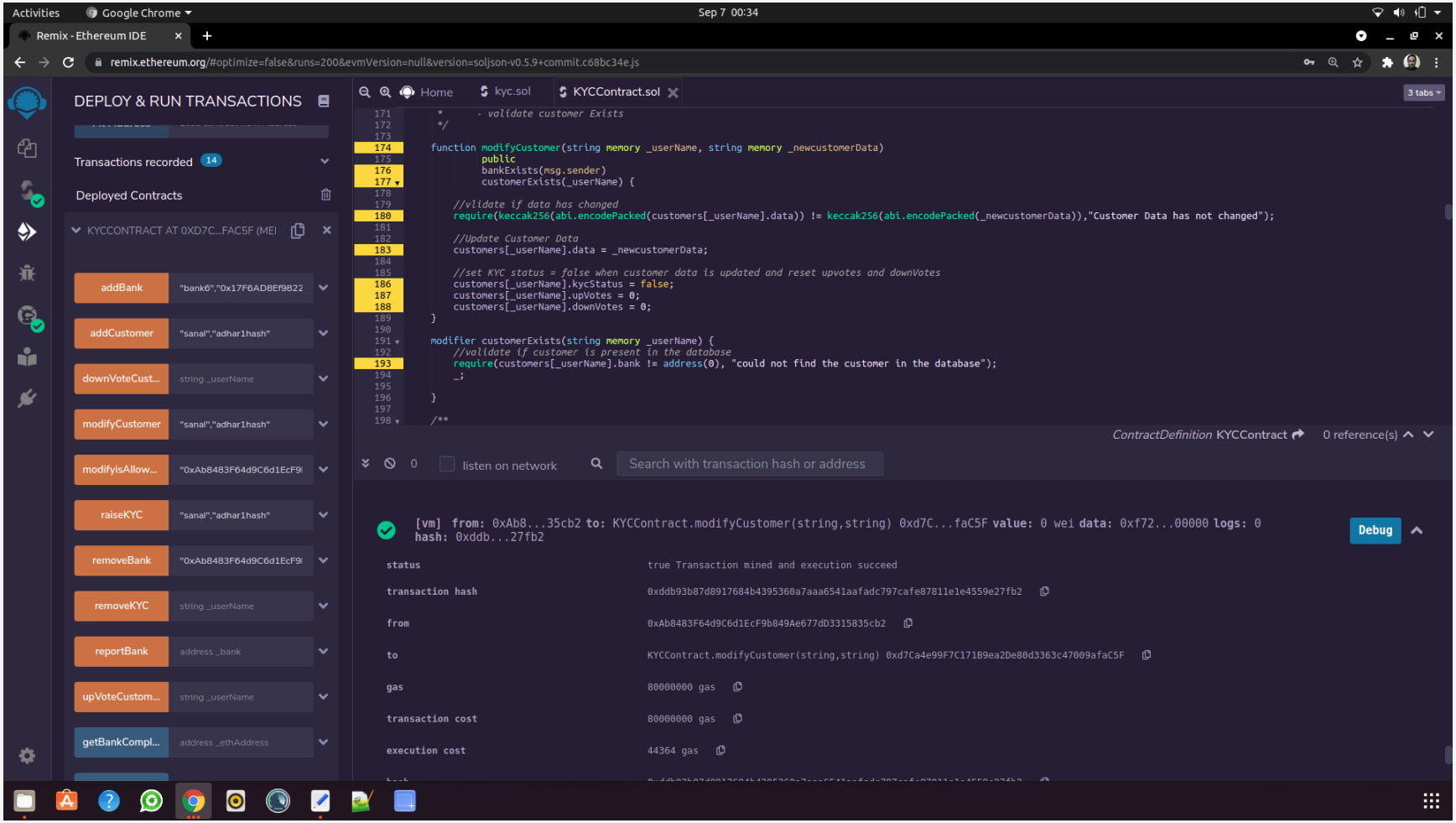
Add customers:



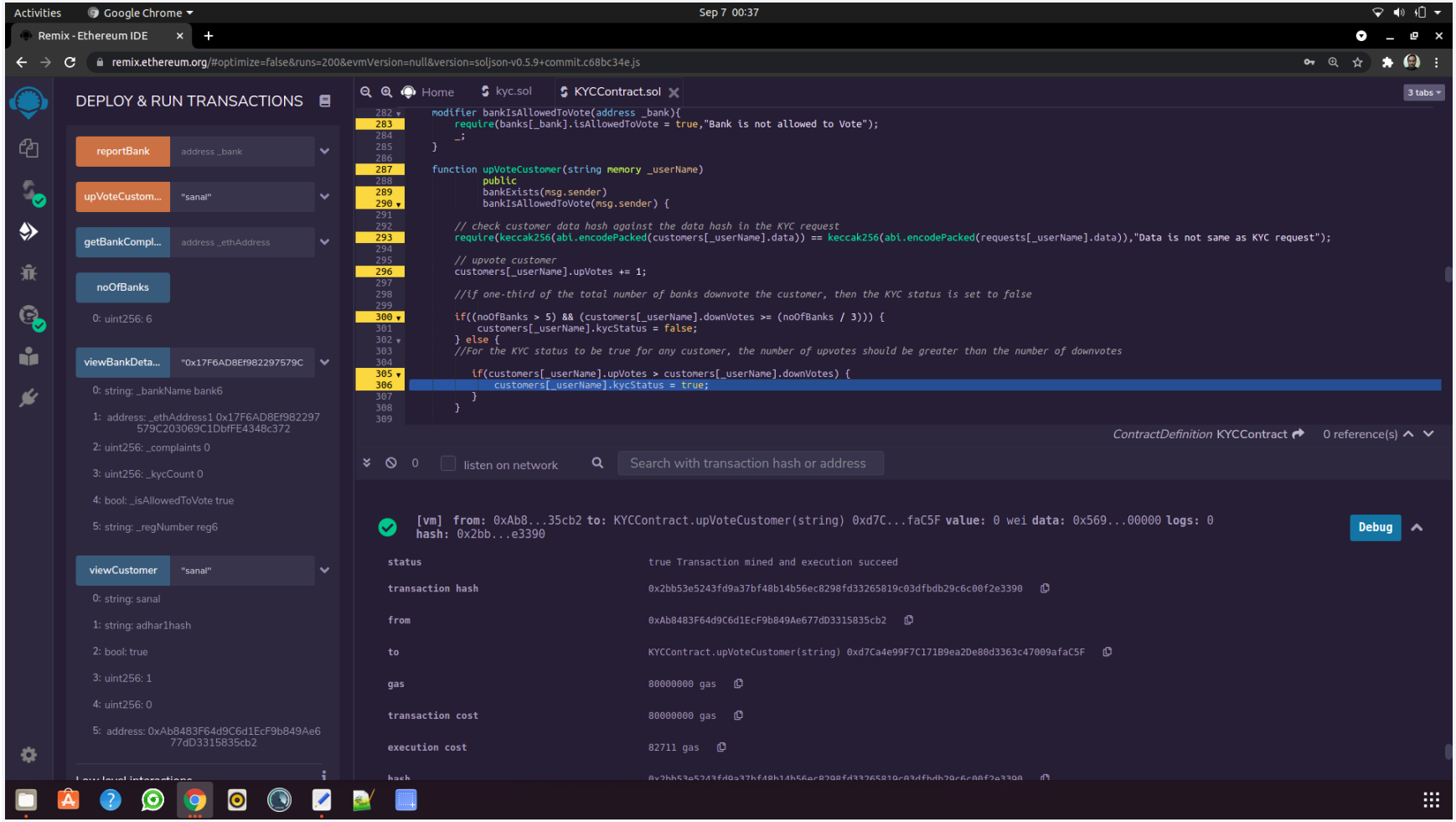
Add KYC request:



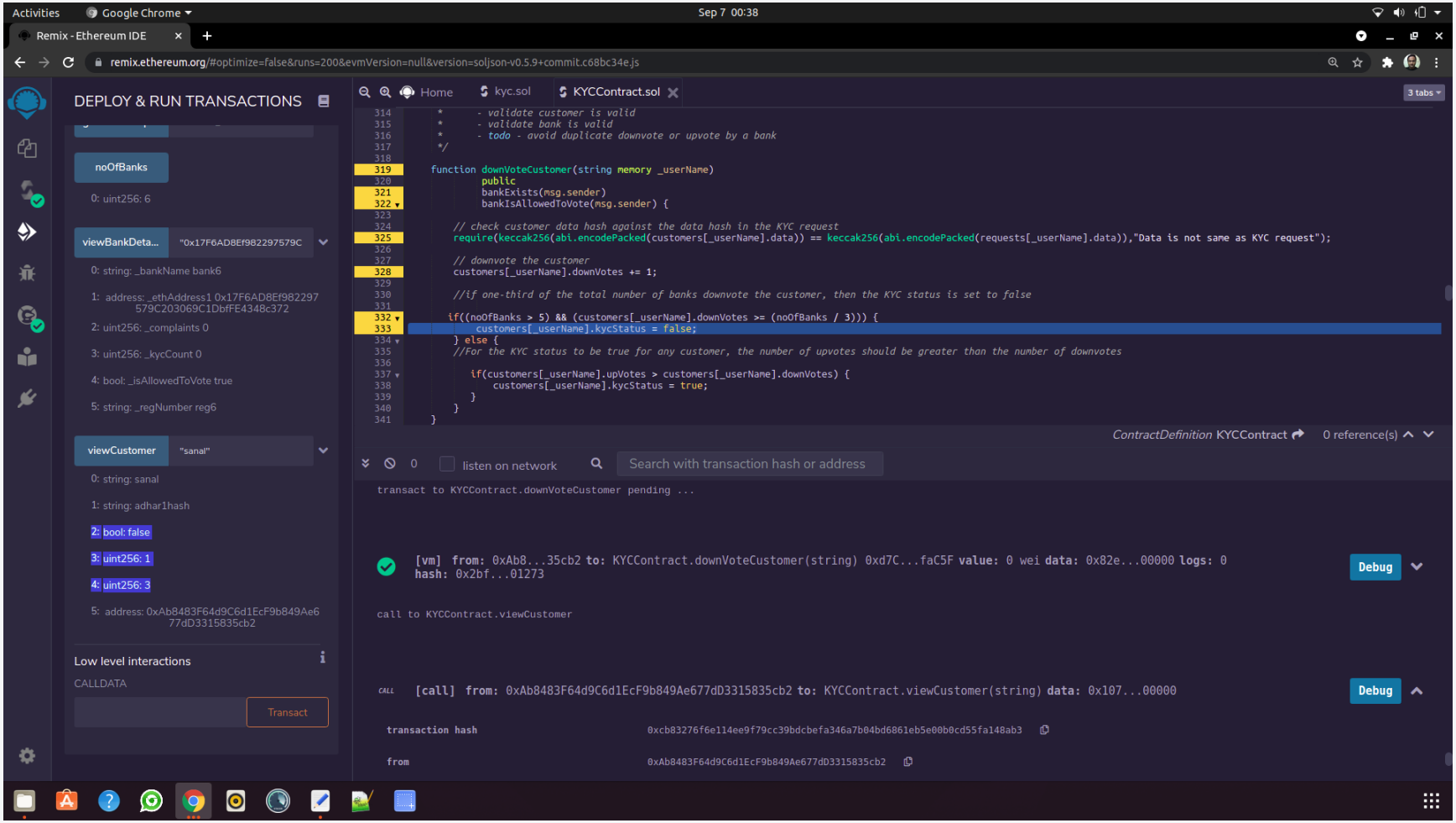
Update Customer:



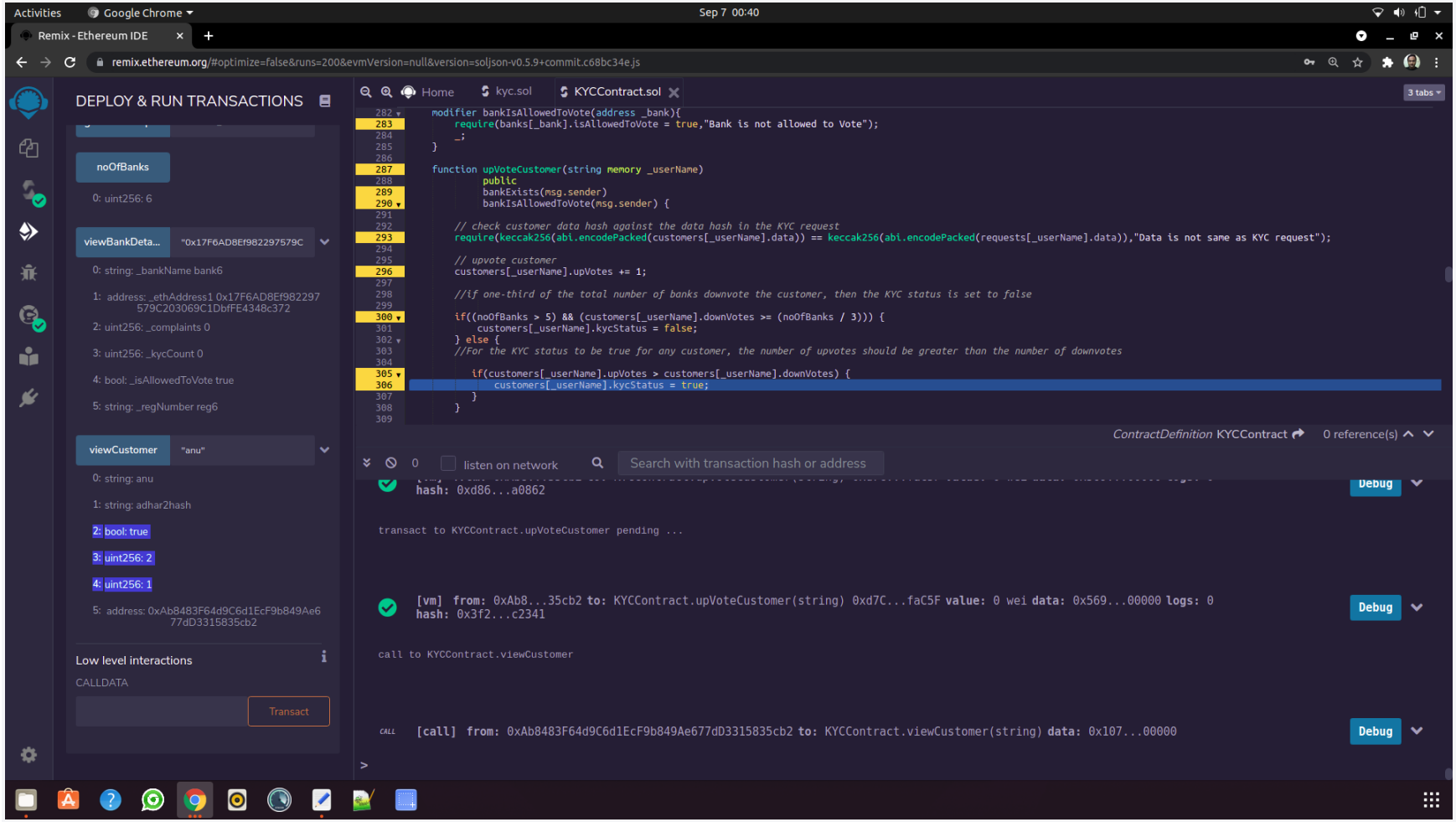
Banks upvote customer: if upvote > downvote -> KYC Status = true



If more than ⅓ rd of banks downvote, customer KYC is false:

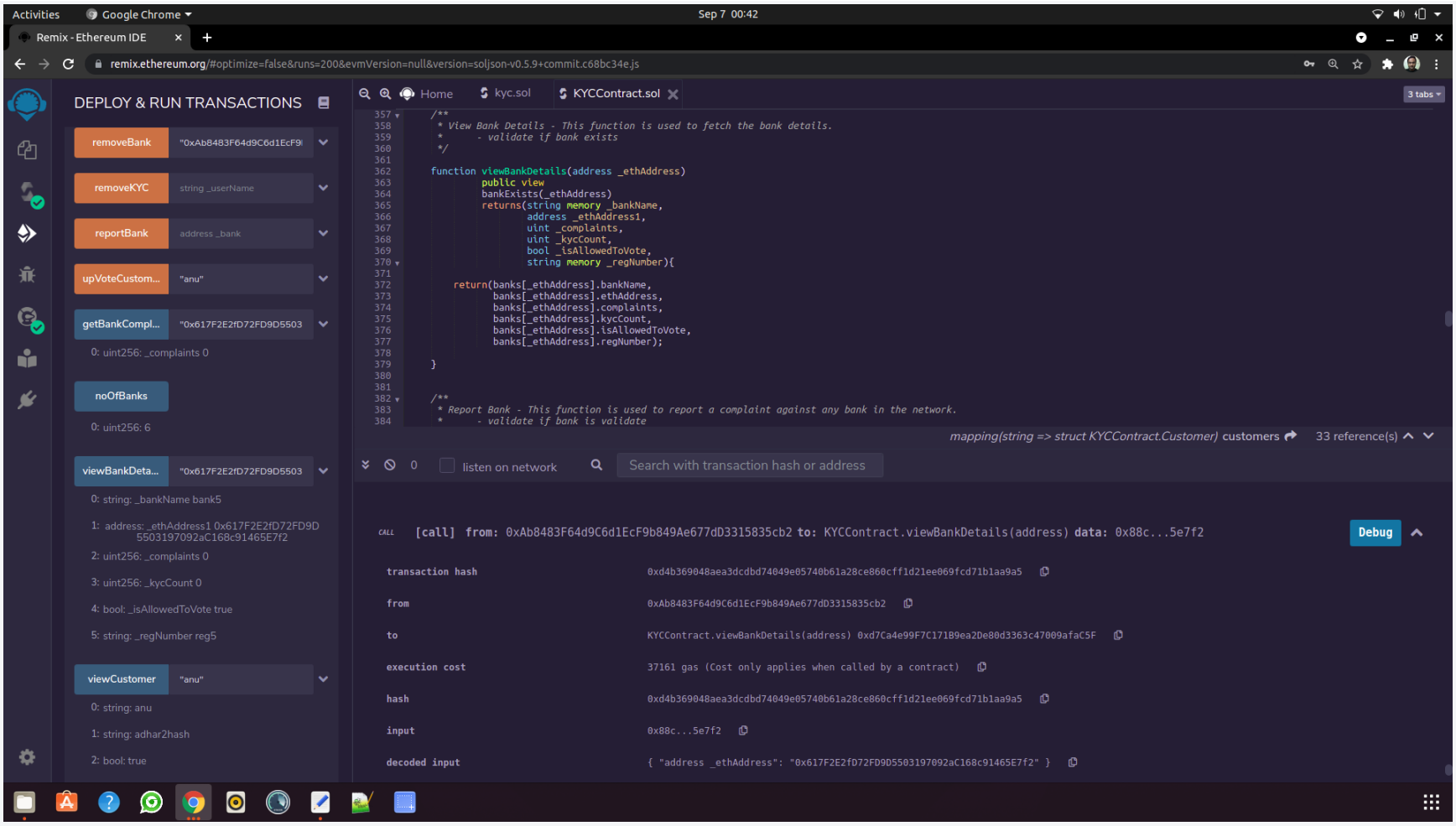


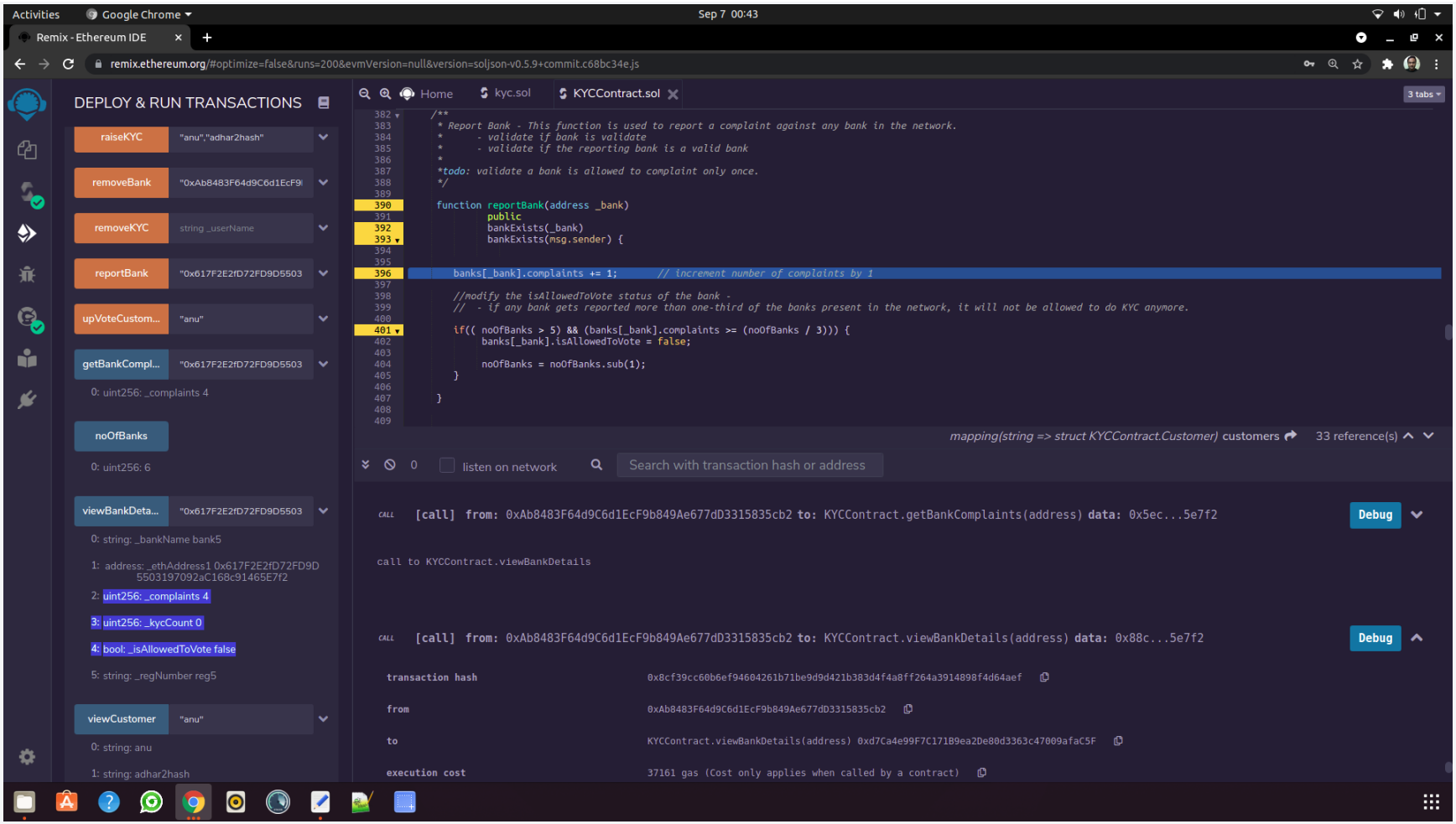
If upvotes is greater than downvotes, KYC is valid:



Banks report another bank:

If more than ⅓ rd of the bank report, banks voting rights are removed.




Activities
Google Chrome
Sep 7 00:43
Remix - Ethereum IDE
remix.ethereum.org/#optimize=false&runs=200&evmVersion=null&version=soljson-v0.5.9+commit.c68bc34e.js
DEPLOY & RUN TRANSACTIONS
raiseKYC
"anu","adhar2hash"
removeBank
"0xAb8483F64d9C6d1EcF9
removeKYC
string _userName
reportBank
"0x617F2E2fD72FD9D5503
upVoteCustom...
"anu"
getBankCompl...
"0x617F2E2fD72FD9D5503
0: uint256: _complaints 4
noOfBanks
0: uint256: 6
viewBankDeta...
"0x617F2E2fD72FD9D5503
0: string: _bankName bank5
1: address: _ethAddress1 0x617F2E2fD72FD9D5503197092aC168c91465E7f2
2: uint256: _complaints 4
3: uint256: _kycCount 0
4: bool: _isAllowedToVote false
5: string: _regNumber reg5
viewCustomer
"anu"
0: string: anu
1: string: adhar2hash
Home
kyc.sol
KYCContract.sol
3 tabs
/**
* Report Bank - This function is used to report a complaint against any bank in the network.
* - validate if bank is validate
* - validate if the reporting bank is a valid bank
*
*todo: validate a bank is allowed to complaint only once.
*/
function reportBank(address _bank)
public
bankExists(_bank)
bankExists(msg.sender) {
banks[_bank].complaints += 1; // increment number of complaints by 1
//modify the isAllowedToVote status of the bank -
// - if any bank gets reported more than one-third of the banks present in the network, it will not be allowed to do KYC anymore.
if((noOfBanks > 5) && (banks[_bank].complaints >= (noOfBanks / 3))) {
banks[_bank].isAllowedToVote = false;
noOfBanks = noOfBanks.sub(1);
}
}
mapping(string => struct KYCContract.Customer) customers
33 reference(s)
0
listen on network
Search with transaction hash or address
call [call] from: 0xAb8483F64d9C6d1EcF9b849Ae677dD3315835cb2 to: KYCContract.getBankComplaints(address) data: 0x5ec...5e7f2
Debug
call to KYCContract.viewBankDetails
call [call] from: 0xAb8483F64d9C6d1EcF9b849Ae677dD3315835cb2 to: KYCContract.viewBankDetails(address) data: 0x88c...5e7f2
Debug
transaction hash 0x8cf39cc60b6ef94604261b71be9d9d421b383d4f4a8ff264a3914898f4d64aef
from 0xAb8483F64d9C6d1EcF9b849Ae677dD3315835cb2
to KYCContract.viewBankDetails(address) 0xd7Ca4e99F7C171B9ea2De80d3363c47009afaC5F
execution cost 37161 gas (Cost only applies when called by a contract)