

# Decentralized KYC Verification Process for Banks

## Project 1

### DESCRIPTION

Central Bank and other government banks face issues in tracking money laundering activities that are used for terrorism and other crimes. It is a threat to national security and is also adversely affecting the economy.

#### Background of the problem statement:

KYC (Know Your Customer) is a service provided by financial institutions such as banks. There are both public and private sector banks managed by a central bank. These banks are banned by the central bank from adding any new customer and do any more customer KYCs as they see suspicious activities that need to be

sorted out first. Despite this, the banks add new customers and do the KYC in the background.

An immutable solution is needed where the central bank maintains a list of all the banks and tracks which banks are allowed to add new customers and perform KYC. It can also track which customer KYC is completed or pending along with customer details.

Banks can also add the new customer if allowed and do the KYC of the customers.

#### Features of the application:

##### 1. Add new bank to Blockchain ledger:

This function is used by the admin to add a new bank to the KYC Contract. This function can be called by admin only.

This function takes the below input parameters:

- bankName of string type: The name of the bank
- address of address type: The unique Ethereum address of the bank

## 2. Add New customer to the bank:

This function will add a customer to the customer list. This function takes the below input parameters:

- custName of string type: The name of the customer
- custData of string type: Customer supporting data such as address and mobile number

## 3. Check KYC status of existing bank customers:

This function is used to fetch customer KYC status from the smart contract. If true, then the customer is verified. This function takes the below input parameter:

- custName of string type: The name of the customer for whom KYC is to be done

Output: Return the KYC status, either true or false.

#### 4. Perform the KYC of the customer and update the status:

This function is used to add the KYC request to the requests list. If a bank is in banned status then the bank won't be allowed to add requests for any customer. This function takes the below input parameter:

- custName of string type: The name of the customer for whom KYC is to be done

#### 5. Block bank to add any new customer:

This function can only be used by the admin to block any bank from adding any new customer. This function takes the below input parameter:

- add of address type: The unique Ethereum address of the bank

#### 6. Block bank to do KYC of the customers:

This function can only be used by the admin to change the status of KYC permission of any of the banks at any point of time. This function takes the below input parameter:

- add of address type: The unique Ethereum address of the bank

7. Allow the bank to add new customers which was banned earlier:

This function can only be used by the admin to allow any bank to add any new customer. This function takes the below input parameter:

- add of address type: The unique Ethereum address of the bank

8. Allow the bank to perform customer KYC which was banned earlier:

This function can only be used by the admin to change the status of KYC Permission of any of the banks at any point of time. This function takes the below input parameter:

- add of address type: Unique Ethereum address of the bank

## 9. View customer data:

This function allows a bank to view details of a customer. This function takes the below input parameter:

- custName of string type: The name of the customer

### Recommended technologies:

1. Smart Contract development: Solidity
2. IDE tool: Remix
3. Blockchain: Ethereum
4. Server: Ganache Blockchain