**Next Generation Compliance De-Centralized Application:**

Our DApp consists of a private block-chain network based on Ethereum. There will be two types of smart contract in the network. A smart contract which stores identity and tracking information about the customers. We call this smart contract KYC master. And a number of smart contracts that read and write to KYC master. We call this type of smart contract KYC clients. As obvious as it is, there will be one KYC master and several KYC clients in our network.

When the network is created for the first time, only KYC master will exist. KYC clients are parties like financial institutions, legal firms, government agencies that are interested to participate in our network. A KYC client can join or leave the network any time they want. Upon joining the network a KYC client has to pay a specific joining fee to the network. The joining fee will be kept by the KYC master as balance of individual KYC client. We explain why this fee is useful in the latter part of the write-up.

A situation comes where one of our KYC clients wants to on-board a customer. Let's call the KYC client 'CL' and the customer 'C'. The most important part of the on-boarding process is completing the KYC process of C. It's highly likely that the KYC process of C was already completed by some other KYC client in our network. So CL sends the unique identifier (Aadhar No, SSN, etc) of C to KYC master and asks if details are avilable. Let's consider that KYC master replies that no information is available for C. So CL decides to initiate the KYC process for C.

Once the process is completed, CL posts back the details of C to the KYC master. In a future date when some other KYC client requests KYC details of C, the KYC master instantly returns the details. Since CL had originally invested money and time in completing the KYC process for C, the KYC master rewards CL. This is done by deducting a small amount from the balance of the KYC client and adding it to CL's balance. This way, over time CL could recover part of the C's KYC expenditure.

Over the course of time C might build relationship with other KYC clients in our network. Hence other KYC clients would also contribute to the KYC information of C. We call an individual contribution by a KYC client to KYC information of C as an 'Event'. We classify Event into broadly three types :-

1) Event-Normal

2) Event-Change

3) Event-Alert

We now explain how our network responds to each event type.

**Event-Normal:**

This is a FYI type of events occurring in the network. For example C has started a relationship with another KYC client in the network. Since this is just a FYI this can be directly added to the KYC information of C. The network normally does not reward for this type of event.

**Event-Change:**

Customer information are subject to change over time. For example, let's say C decides to change their legal name. They go to a law firm who happens to be a KYC client in our network. Let's call this law firm 'CL1'. CL1 completes the process to change the legal name of C and updates the KYC master. KYC master notifies all KYC clients and calls for a voting to acknowledge this change. A majority (Let's say 66%, we can configure this) of the KYC clients have to respond with an 'Yes' for this event to be added to the KYC information of C. If the majority of the KYC clients say 'Yes', the network rewards CL1 for their effort.

**Event-Alert:**

Some customers start a good relationship but become evil afterwards. Let's say one of our KYC clients detects C is likely involved in a money laundering activity. This is type of event is reported as Event-Alert to KYC master. KYC master immediately notifies all KYC clients. As with Event-Change, Event-Alert would require all KYC clients to vote and acknowledge. Since this is an alert, the majority and reward should be configured like:-

Majority (Event-Alert) > Majority (Event-Change)

Reward (Event-Alert) > Reward (Event-Change)

It could be noted that the network rewards a KYC client for creating a genuine KYC information of C by deducting amount from the balance of the other KYC clients. The KYC master will notify a KYC client if their balance falls below a minimum amount. The KYC master also provides the KYC clients a way to top-up their balance.