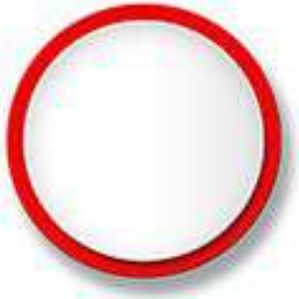
RanchiMall

Architecture for FLO based Applications

Technology



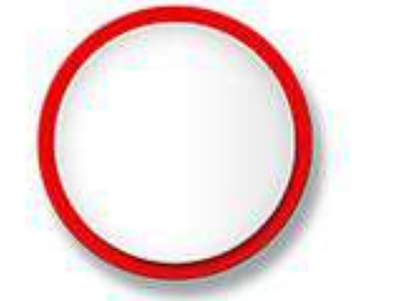
RANCHIMALL

INTRODUCTION

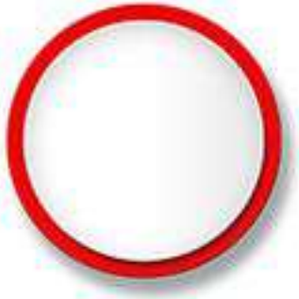
RanchiMall has designed an architecture for FLO based decentralized applications (Dapps).

We call it FLO Standard Operations.

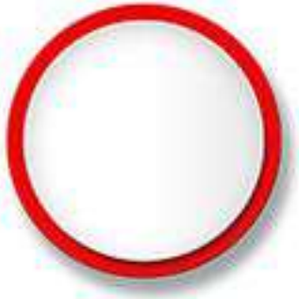
It embeds a philosophy to create Dapps and also the technical know-how needed to enable it.



So lets talk about philosophy first.

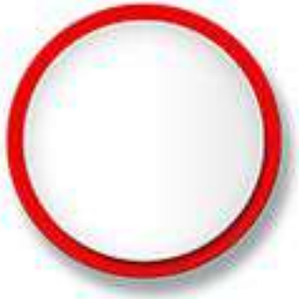


When an application is centralized, the trust in the application gets centralized as well.



A user has no choice whom to trust.
He must trust the centralized server for everything.

There is single point where all trust gets concentrated.

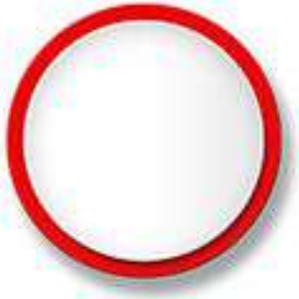


The user needs to trust the central server for:

1. Safe keeping of his password related information
2. Safe upkeep of his personal data
3. Safe upkeep of credit card and payment information

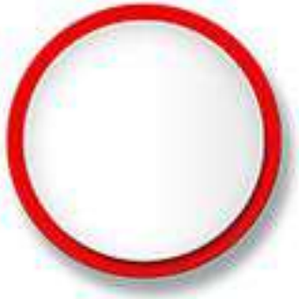
The user also needs to trust the server that it is applying the policies fairly to all users.

In case of distributed applications, the user fundamentally trusts one blockchain address.



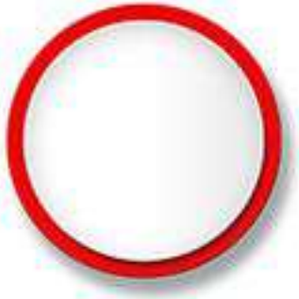
There is no way any system can be designed such that it can take away the need to trust completely.

Someone must be trusted.



In case of Bitcoin everyone trusts the genesis block of Bitcoin Blockchain.

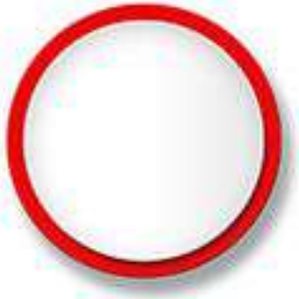
That genesis block is hardcoded in the software.



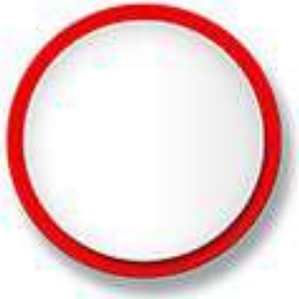
The Bitcoin software can be run with a different genesis block as well.

Everything in ecosystem will remain the same.

But Bitcoin balances of every wallet will change dramatically.

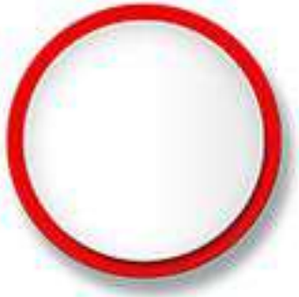


Compared to trusting a centralized software, trusting a blockchain address is much more trustworthy.

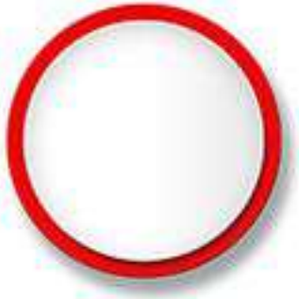


Unlike in centralized servers, information written in blockchain can never change.

And that information stays in blockchain forever.



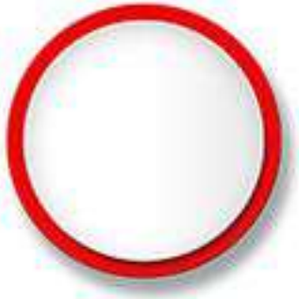
So when a user trusts a blockchain address, he has the full track record permanently of what that address asked users to trust.



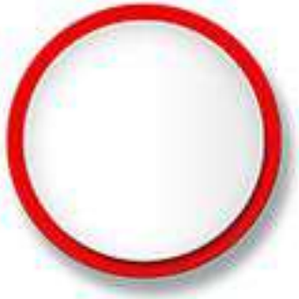
If the blockchain address issues contradictory statements, it will be self evident to everyone.

A centralized server can simply delete the inconvenient facts.

So it is always safer to trust a blockchain address rather than a centralized server.

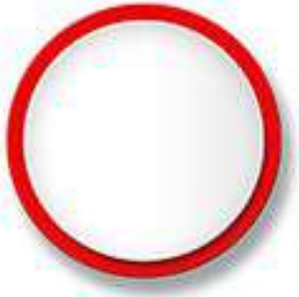


In RanchiMall FLO architecture,
we name the blockchain address
to be trusted as adminID.

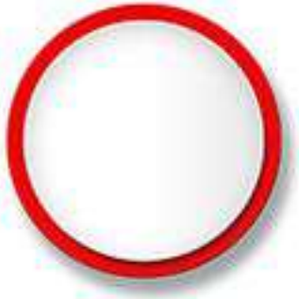


Since the adminID is all-powerful,
it should be only used rarely.

Otherwise the private key of adminID
can be compromised.



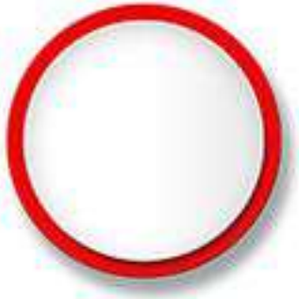
So we create another role called subadminID which is another FLO ID.



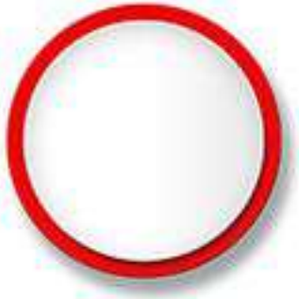
SubadminID can perform regular administrative functions for the system.

AdminID can nominate several subadminIDs by commanding it in the blockchain under its FLO ID.

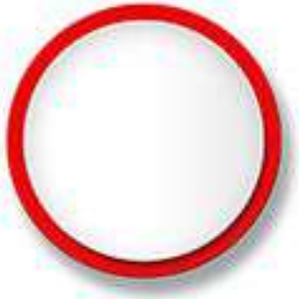
So then the job of adminID is just to nominate subadminIDs, and subadminIDs will do the actual administration.



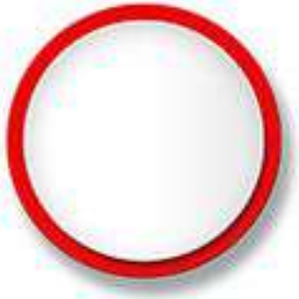
Each distributed application can create its own rules what the subadminIDs can do.



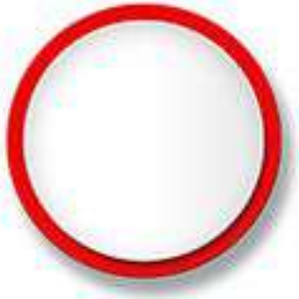
So a normal browser client first checks the adminID in blockchain, finds all active subadmins, and then downloads all messages from the subadmins from the cloud.



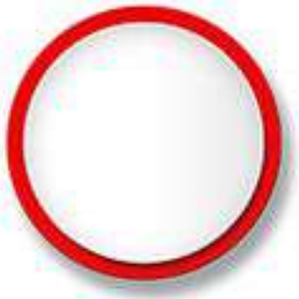
The browser then proceeds to execute the commands issued by subadmins.



Another aspect of FLO Application Design is data modeling.
It is said, if you want to do a clean design, fix the data first.



Since applications can have data stored in blockchain as well as cloud, system designers have to decide what data will go in blockchain, and what data will go in cloud.

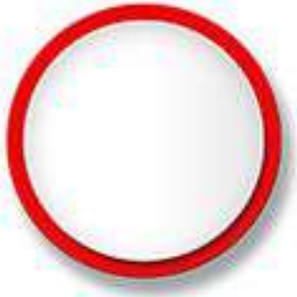


RanchiMall

The cloud offers two fundamental kinds of data storage.

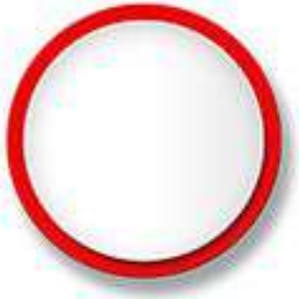
General Data: This type of data is used to collect information from general users.

Object Data: Once general data is received from general users, subadmins can store it in Object Data. Object data is more easy manipulated, but for safety reasons only subadmins should manipulate it.



RANCHIMALL
INTRODUCTION

All users can in theory create both general data and object data, but safe application practices would demand only subadmins manipulate object data directly.

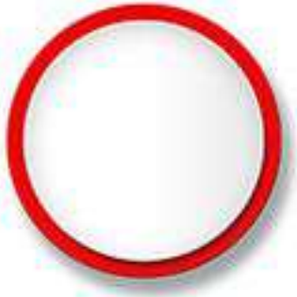


In application design process,

1. Start with figuring out the core functionalities first.

2. Do the data modeling as soon as functionality is known.

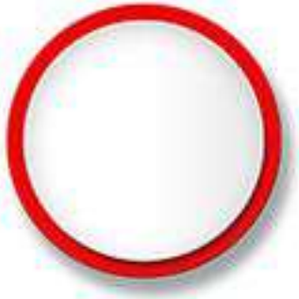
3. Always start the actual application construction using UI (User Interface) first approach.



UI First Approach

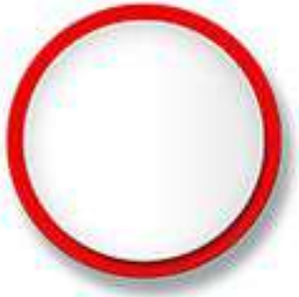
RanchiMall provides variety of different standard UI layouts to organize the functionality.

Always begin the actual construction of the application by picking one of standard UI layouts, and start hooking it up with right data pieces.



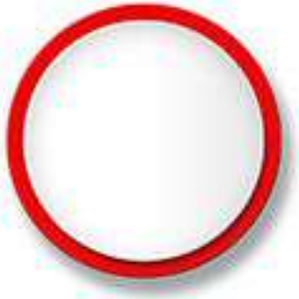
Standard UI components

RanchiMall also provides standard UI components that makes it easy to integrate any UI element that also needs JavaScript.



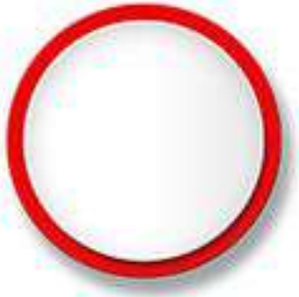
For example most comment items like JavaScript enabled input boxes, carousels, user notification elements are provided as a standard package with automatically integrated CSS and JavaScript.

This makes creating dynamic UI very easy.



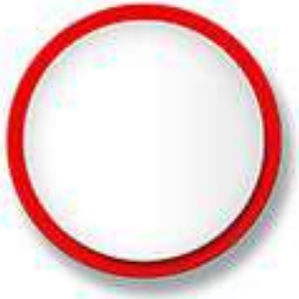
After Data modeling and UI is settled, then comes creation of actual JavaScript functions.

This is the core of code construction.



How to create JavaScript Functionality

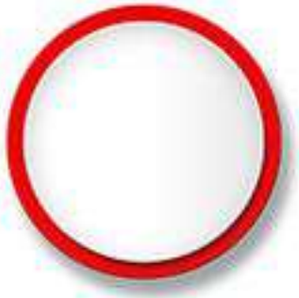
1. Create the most core functions first.
2. Link it to UI and data elements.
3. Then start creating secondary functions, and doing UI/Data linkups.



So the approach till now

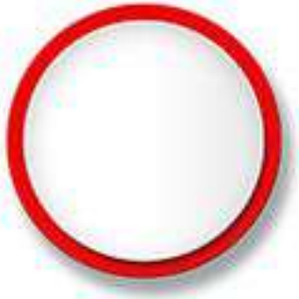
Design phase

1. Figure out the functionality first.
2. Determine role of adminID and subadminIDs in terms of those functionalities.
3. Create the data modeling.



Construction Phase:

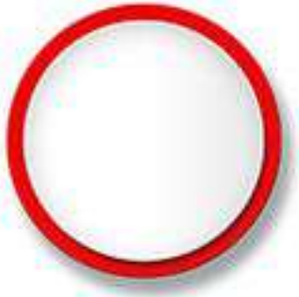
1. UI first approach: Organize the functionality using one of standard layouts.
2. Hook up UI with data pieces as per data modeling.
3. Create the JavaScript functions needed to execute the functionality.



Distributed Data

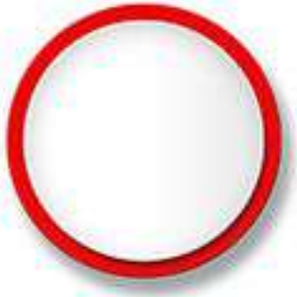
Due to unique nature of the cloud, we get the advantage of distributed data design.

What this means is multiple applications can access the data objects independently.



Once you create the data model in the cloud, any application can access it.

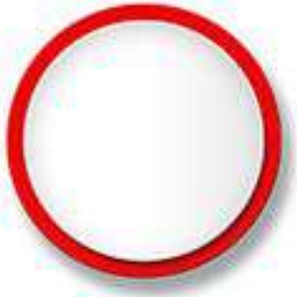
This is unlike centralized applications where one database is owned by only one code base, and for security and integrity reasons, it cannot be shared with other applications.



In distributed application design, any data object can be accessed by any application independently as we do not need password based authentication.

Just sign your data digitally, and store it on the cloud. Then that data cannot be tampered with.

If privacy is required, then encrypt the data with the intended recipient FLO ID public key.



Standard Operations

RanchiMall provides the following standard operations to simplify application creation:

1. FLO Globals for system variables and data objects users must configure.

2. FLO Crypto Operations for digital signatures and encryption.

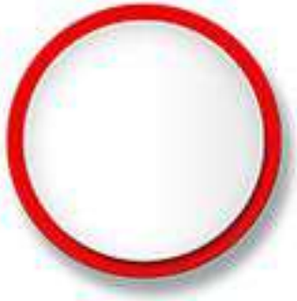
RanchiMall

3. FLO Blockchain API Operations for reading and writing into FLO Blockchain.

4. Compact IndexedDB Operations for easy access to local IndexedDB.

5. FLO Cloud API Operations for reading and writing data to cloud.

6. FLO Decentralized app (Dapp) Operations for most common Dapp operations.



RANCHIMALL
INTRODUCTION