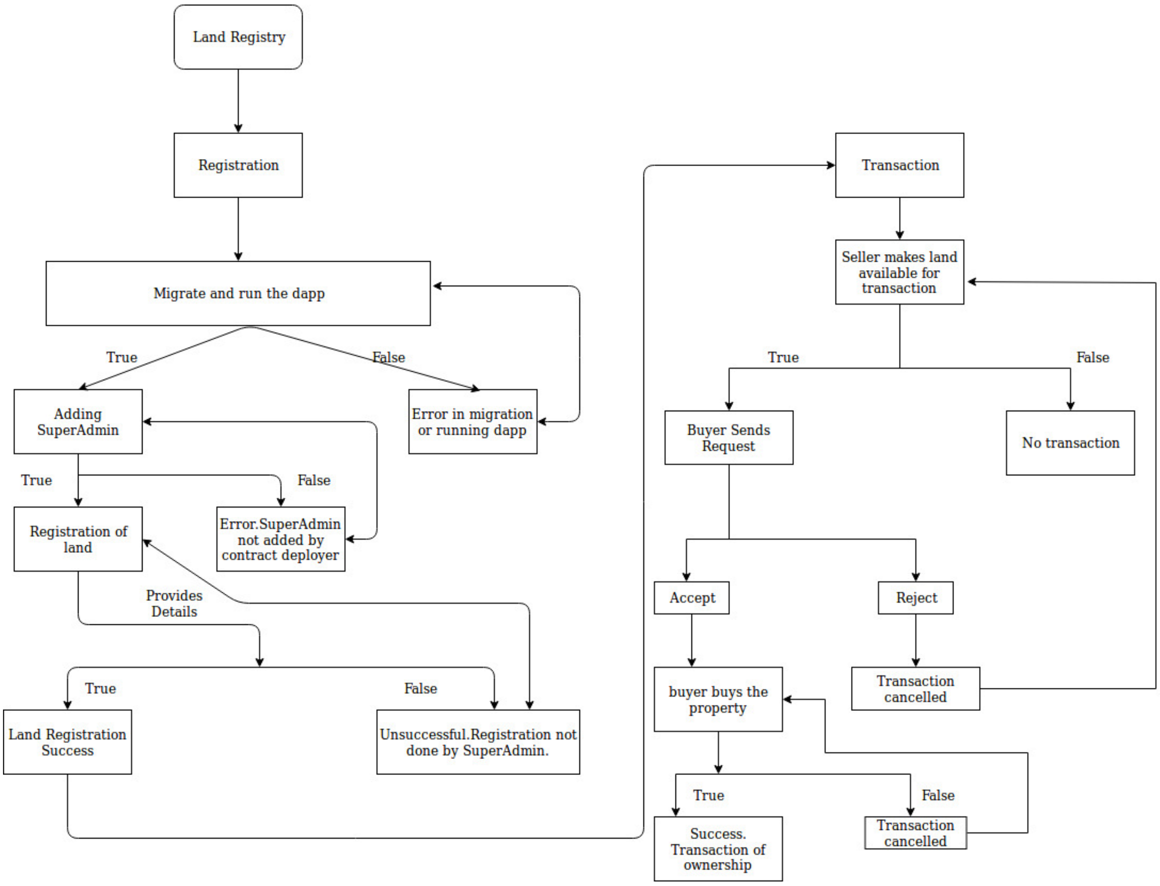
LAND REGISTRY-DOCUMENTATION

CODE FLOW :



FUNCTIONS() :

1. addSuperAdmin() : Used for adding the super admin. This function can only be called by the owner of contract(person who deployed the contract). Address (ethereum) of super admin and the village to which the super admin is assigned for is provided as input. Each super admin's address is mapped to the corresponding village.

2. Registration() : Used for registering a land. Details of land such as state, district,village and survey number along with the owner's ethereum address, market value of the land and property ID is provided as the input. The details are registered mapped to the property ID. This function can only be called by the super admins and the registration of a land can only be done by the super admin of the village to which the land belongs to.

3. landInfoOwner() : Function to view the details of land for the owner. Input is property ID and returns state, district, village, survey number,

availability to buy, address of the person who requested the land to buy, request status(accept,reject or pending) as the output.

4. landInfoUser() : Function to view the details of land for any user. Input is property ID and output is the current owner of the property,market value, availability of the land to buy, whether there are any requester for the land(address of requester) and request status.

5. computeId() : Used to compute a unique ID for a property. State, district, village and survey number is provided as input and a unique ID is returned as the output.

6. requestToLandOwner() : Used to send a request to buy a property. Property ID is provided as the input and requester's address is sent to the owner.

7. viewAssets() : Function to view the assets of a person. When the function is called, the asset list of the person who called the function is returned.

8. viewRequest() : Function to view the address of the requester for a particular land, if there are any. Input is the property ID and the output is the address of the requester.

9. processRequest() : Used to process the request came to buy the land. This function can only be used by the owner of the land. Property ID and request status(accept or reject) is provided as the input.

10. makeAvailbale() : Function to make a property available to buy. This function is also restricted to the land owner. Property ID is provided as the input. If the address matches the land owner the land is made available to buy.

11. buyProperty() : Function to buy the property which has already been approved to buy by the owner. This function can only be called by the requester whose request has been approved. The property ID is provided as the input and the transaction of ether corresponding to the market value and land tax is taken in exchange for the ownership of the property.

12. removeOwnership() : Function is called inside buyProperty(). Used to remove ownership of the previous owner of land. The property ID and the previous owner address is passed from buyProperty().

13. findId() : Function to find the index of the property in the asset list of the previous owner. This function is used inside removeOwnership(). Property ID and the previous owner address is passed from the removeOwnership().

FRONT-END WORKING :

Even though there are so may functions involved in the contract, the user interface is really optimised and simplified to make the process easier. When the dapp is run, the registration page is availbale by typing "/admin" to the existing address which is "http://localhost:4200". The page gives two fields.

One is to register the properties and the other registers the superadmin.

In the transaction part, when user opens the dapp at “http://localhost:4200”, the page gives you the details of the land owned by the account with which you are signed in in metamask. If there are none then the page will be blank. If there is a property then it shows you the details of the property and a button to make the property availble to buy. Once you press the “make available” button then it automatically calls the function “makeAvailbale()” and button is then disabled. If someone wants to buy the property, they can enter into the “search” link in the title bar. This will give you page where you search for a property. If the property is available to buy then a button will appear to send request which will call the function “requestToLandOwner()”. The seller then will receive the request and 3 additional buttons will appear in his/her property details. One is just to show the requester address which will show you nothing on click. The other two are accept and reject buttons. If the seller rejects the request then if you reload the page, the “make available” button would again be active. If the seller choose to accept the request then the property will be availble for the buyer to buy. Now if the buyer search the land again he/she would have an active button “buy” which if you press will transact the property to the buyer by taking corresponding ether from the account. [*Refer DESIGN_DECISIONS to get more clarity regarding the contract and function calls.*]