**Health Record System using Hyperledger Composer Blockcahin**

* Hyperledger Composer :-

Hyperledger Composer is a tool for building blockchain business networks that make it simple and fast for business owners and developers to create blockchain applications to solve business problems.

The Hyperledger **Composer Playground** provides a user interface for the configuration, deployment and testing of a business network. Advanced **Playground** features permit users to manage the security of the business network, invite participants to business networks and connect to multiple blockchain business networks.

URL :- <https://composer-playground.mybluemix.net>

* Blockchain Network Definition has three core components :-

1. The Model written in .cto files
2. The Business Logic written in .js files
3. The Access Control Logic written in .acl files

(Optional) There is a Query file written in .qry files that help us to query over the persistent DB.

These three components are easy to maintain and they govern our whole business application on the Composer

In a model file, you will be defining the following,

1. A namespace - namespace composers.participants

Namespace composer.healthrecord

2. Resources, which includes

* 1. Participants – Doctors, Patients
  2. Assets – PatientInfo, Appointment, etc.
  3. Transactions- updateContact, addAppointment, sendbill, etc.

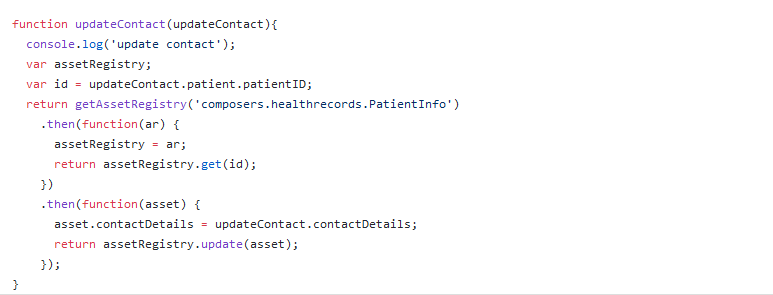
3. Optional import statements that are used to inherit or import other model files.

In Business Logic files, lib\finacial.js or lib\healthrecords.js :

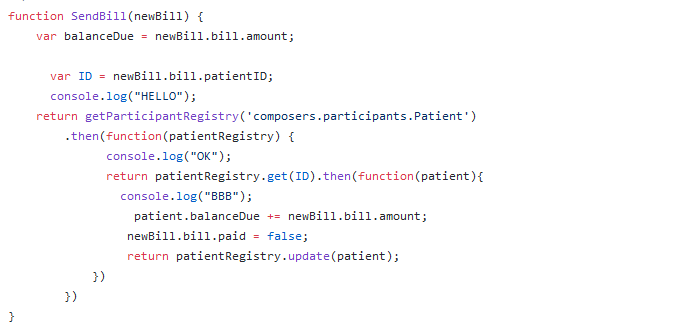
It contains the business logic (smart contract) code, including transaction implementation.

All Transactions behavior is defined in .js files as shown:

updateContact ():



SendBill() :



Like this all the transactions declared in model files are defined in scripts.

What about access control?

Access control is governed by a file in the model called permissions.acl.Access control as one of the primary concepts. Let's have a look at permissions.acl in the **Define** tab of your model. The perishable-network template included an Access Control List (ACL) file that looks like this:

|  |
| --- |
| /\*\*   \* Sample access control list.   \*/  rule Default {      description: "Allow all patients access to all records"      participant: "ANY"      operation: ALL      resource: "composer.finacial.\*"      action: ALLOW  } |

The ACL file contains rules that let you control access to the resources in your blockchain application. Suffice it to say, Hyperledger Composer has you covered when it comes to security.

* Test the business network :-

Notice the Assets and Participants from the model appear on the left side of the screen, but in the center of the screen is a message saying the registry is empty.

To execute the SetupDemo transaction, click the **Submit Transaction** button

Steps to follow the complete the transactions:

1. Click the **Submit Transaction** button
2. Make sure **SendBill** appears in the **Transaction Type** drop-down.
3. Change the "patientId" in the **JSON Data Preview** window.
4. Make sure the patientId is set
5. Click **Submit**.
6. Repeat for the remaining transactions.

In this way one can make the transactions and which will be reflected in All Transactions.

All Transaction button will give you the record of all transactions done in the payground.