HYPERLEDGER

IMC Institute

Developing Business Blockchain Applications on Hyperledger

May 2018

Dr.Thanachart Numnonda

IMC Institute
thanachart@imcinstitute.com

HYPERLEDGER

IMC Institute

Launch an Ubuntu virtual server Using Google Cloud Platform

(You can skip this part if you already have an Ubuntu server)

Thanachart Numnonda, thanachart@imcinstitute.c

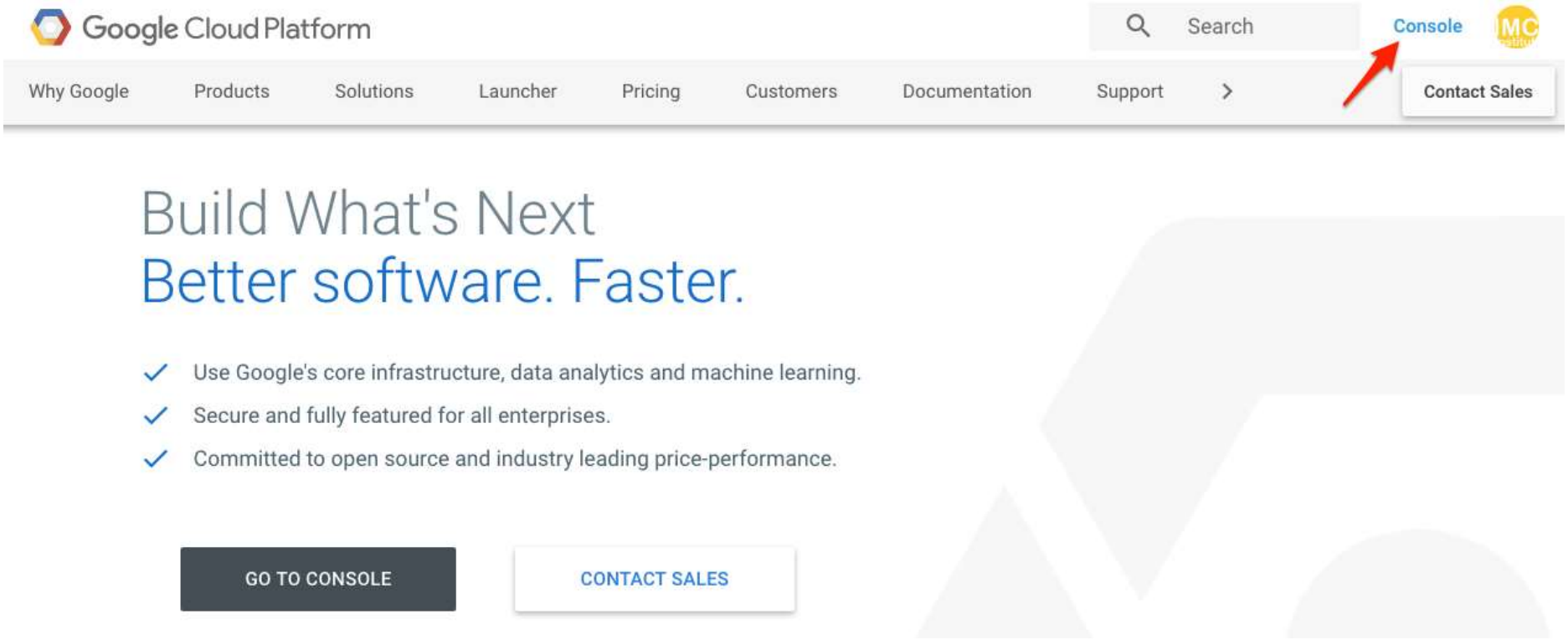
Launch Google Cloud Virtual Server

In this lab, we will use a GCP's compute engine as our server v

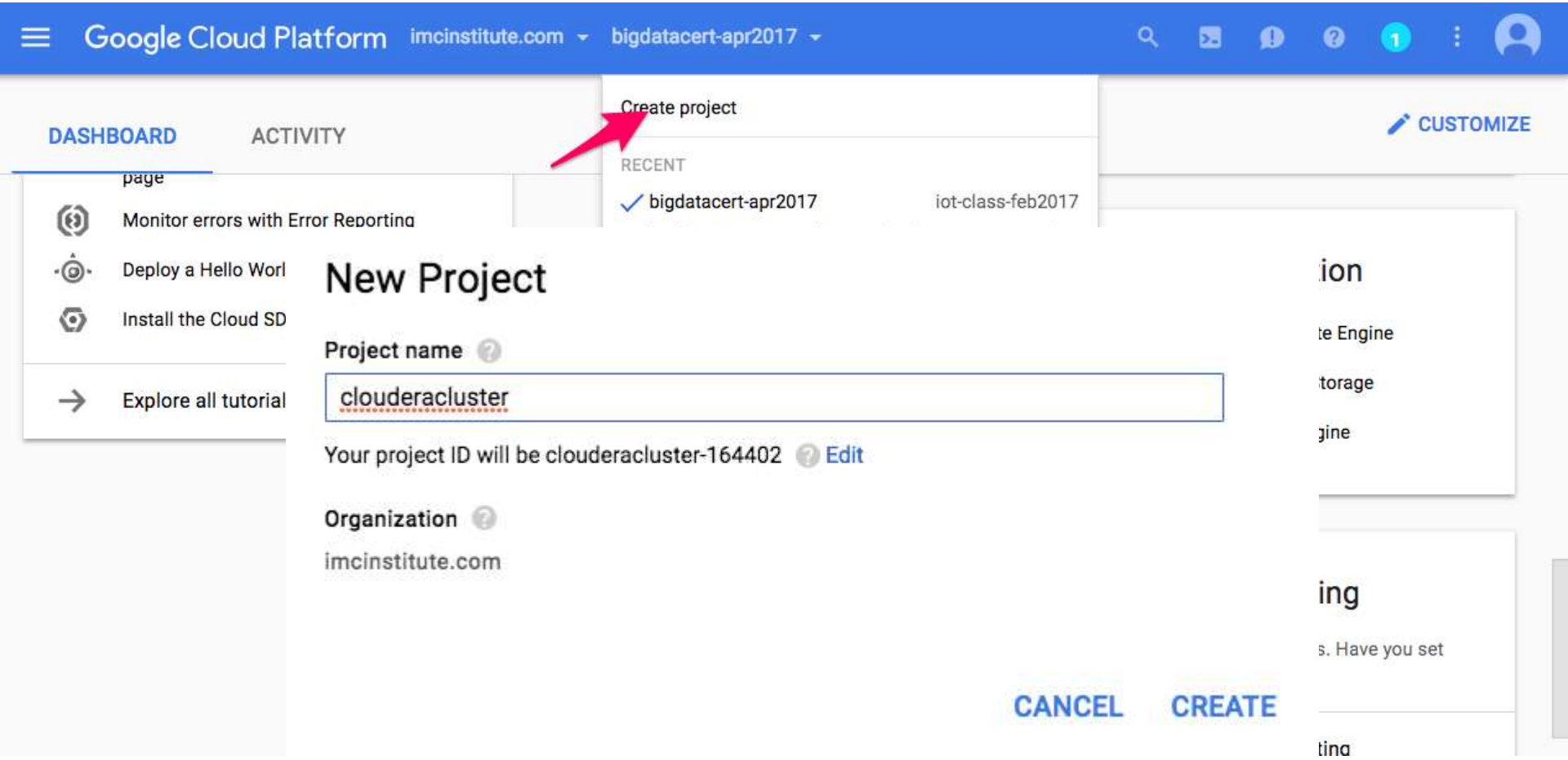
- Ubuntu Server 14.04 LTS
- 2 vCPU, 7.5 GB memory
- 50 GB SSD

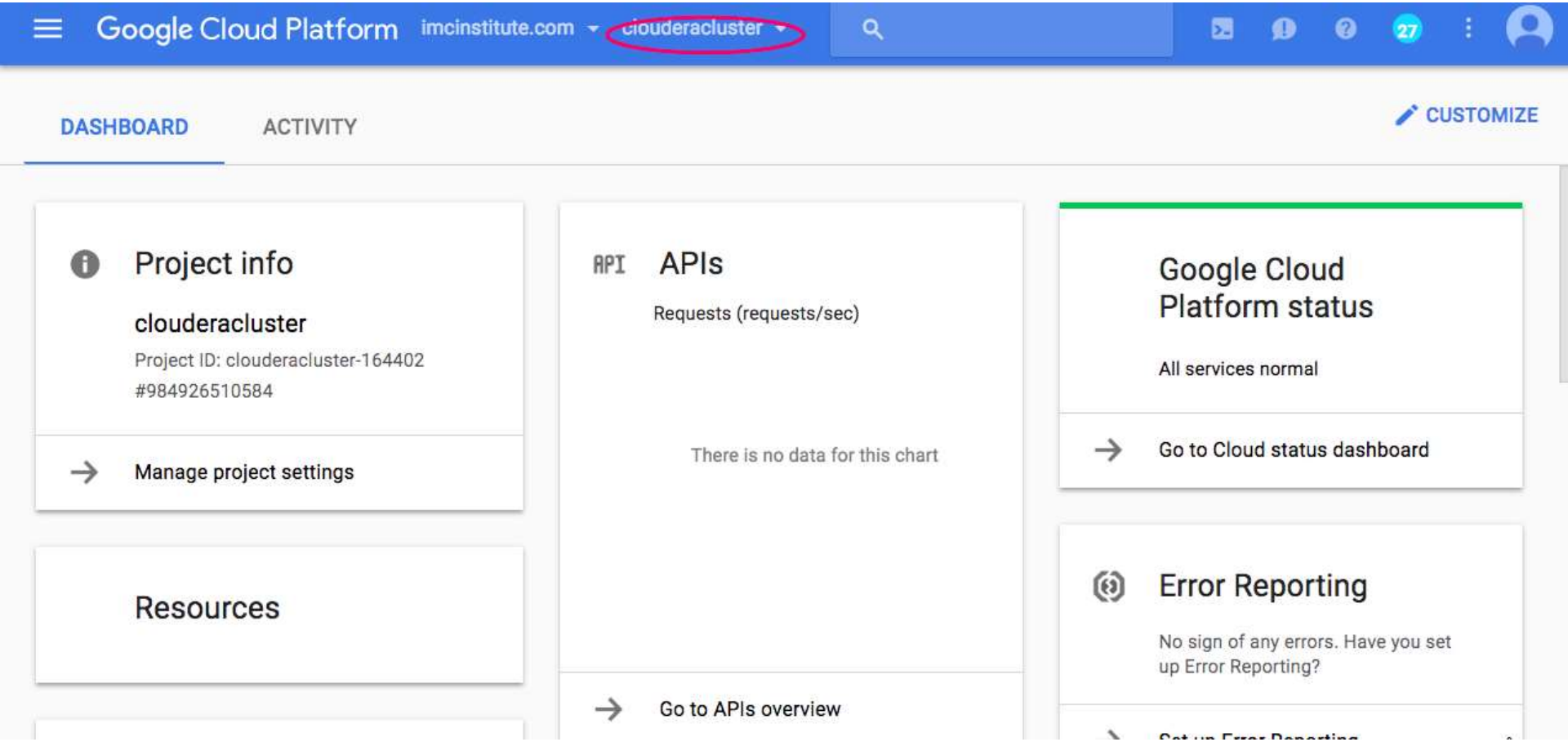
HYPERLEDGER

cloud.google.com

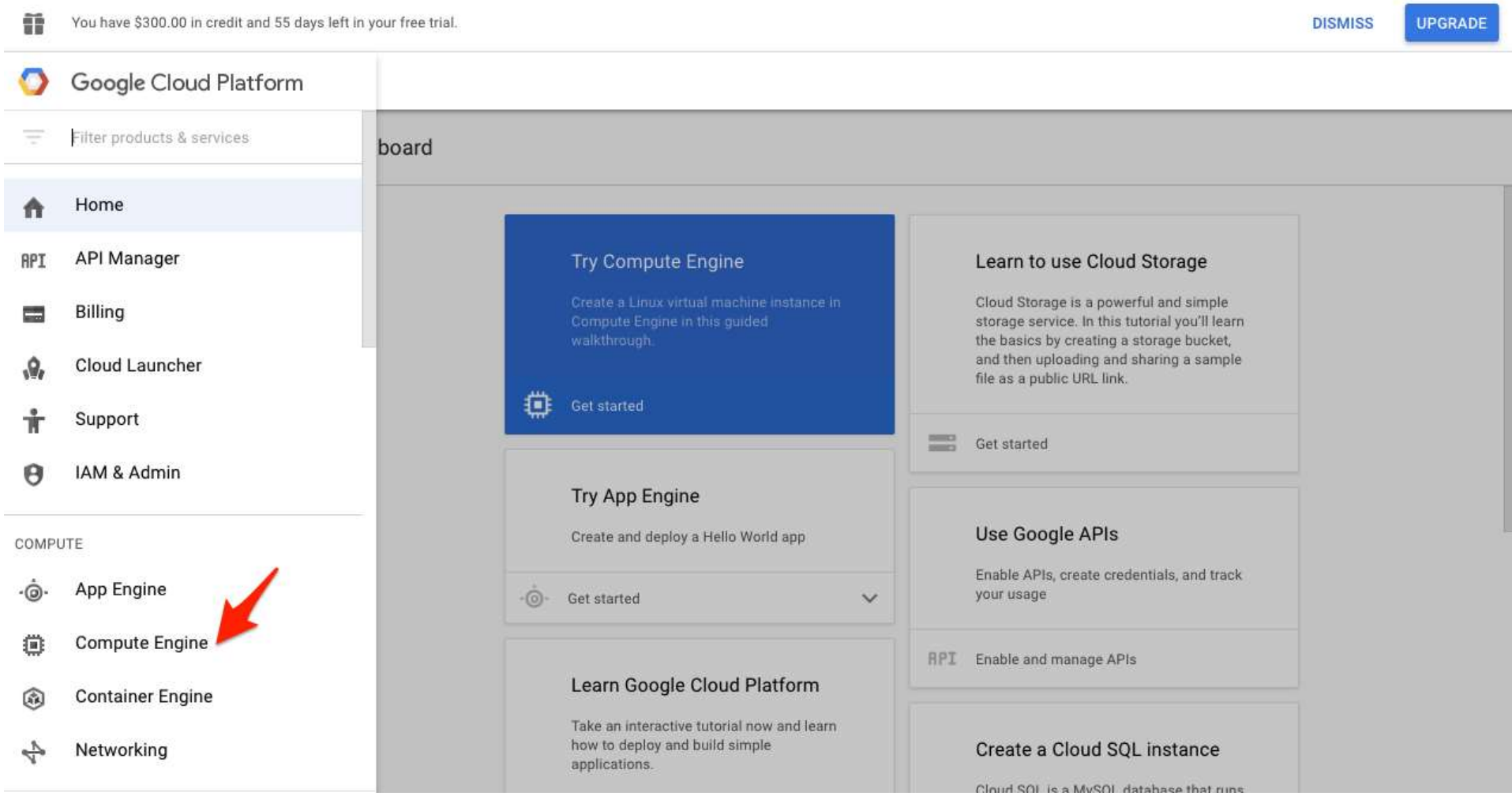


Create Google Cloud Project

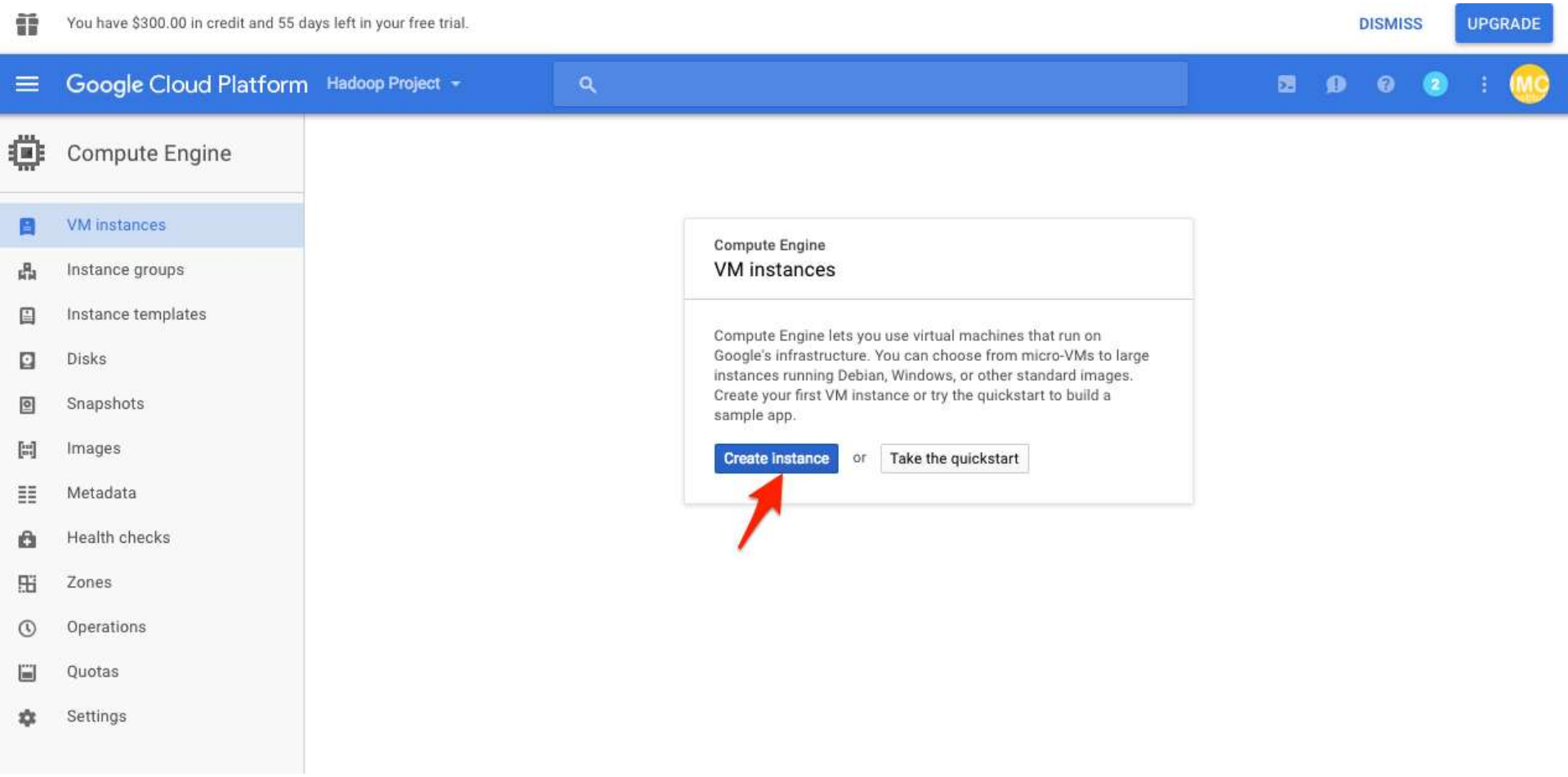



Google Cloud Platform
imcinstitute.com
clouderacluster
27
DASHBOARD
ACTIVITY
CUSTOMIZE
Project info
clouderacluster
Project ID: clouderacluster-164402
#984926510584
Manage project settings
Resources
API
APIs
Requests (requests/sec)
There is no data for this chart
Go to APIs overview
Google Cloud Platform status
All services normal
Go to Cloud status dashboard
Error Reporting
No sign of any errors. Have you set up Error Reporting?

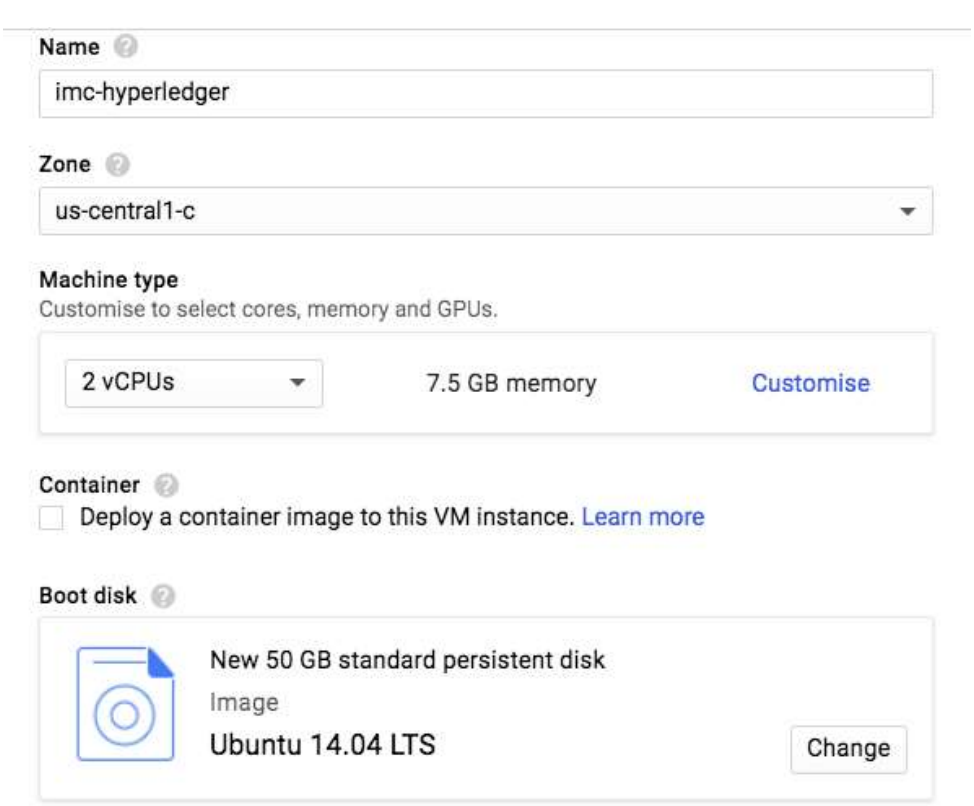
Select Compute Engine

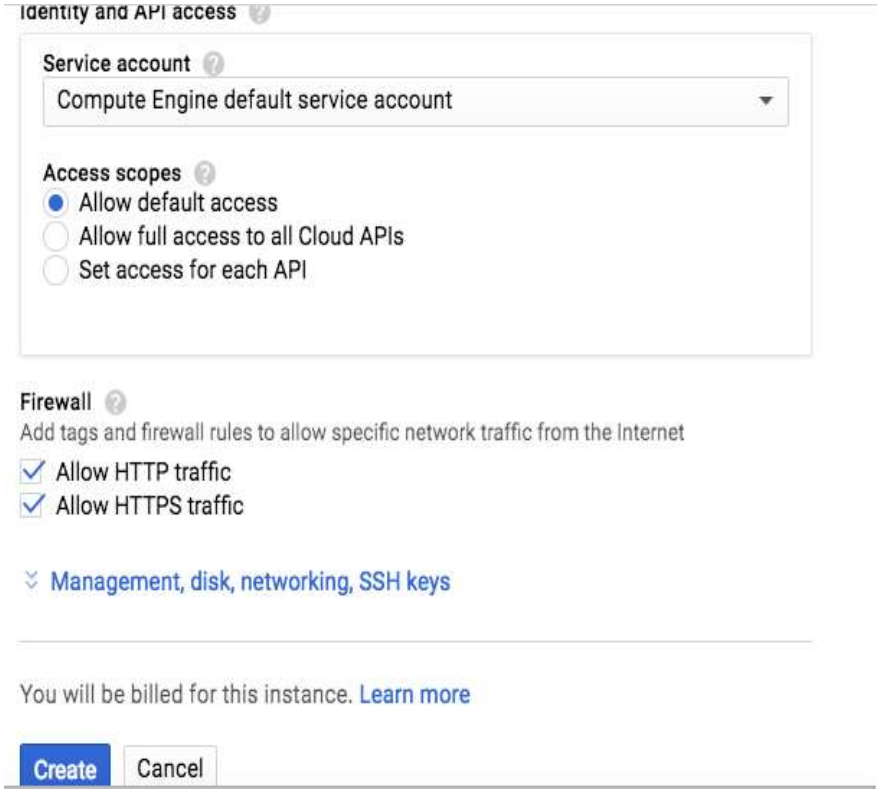


Select Create Instance

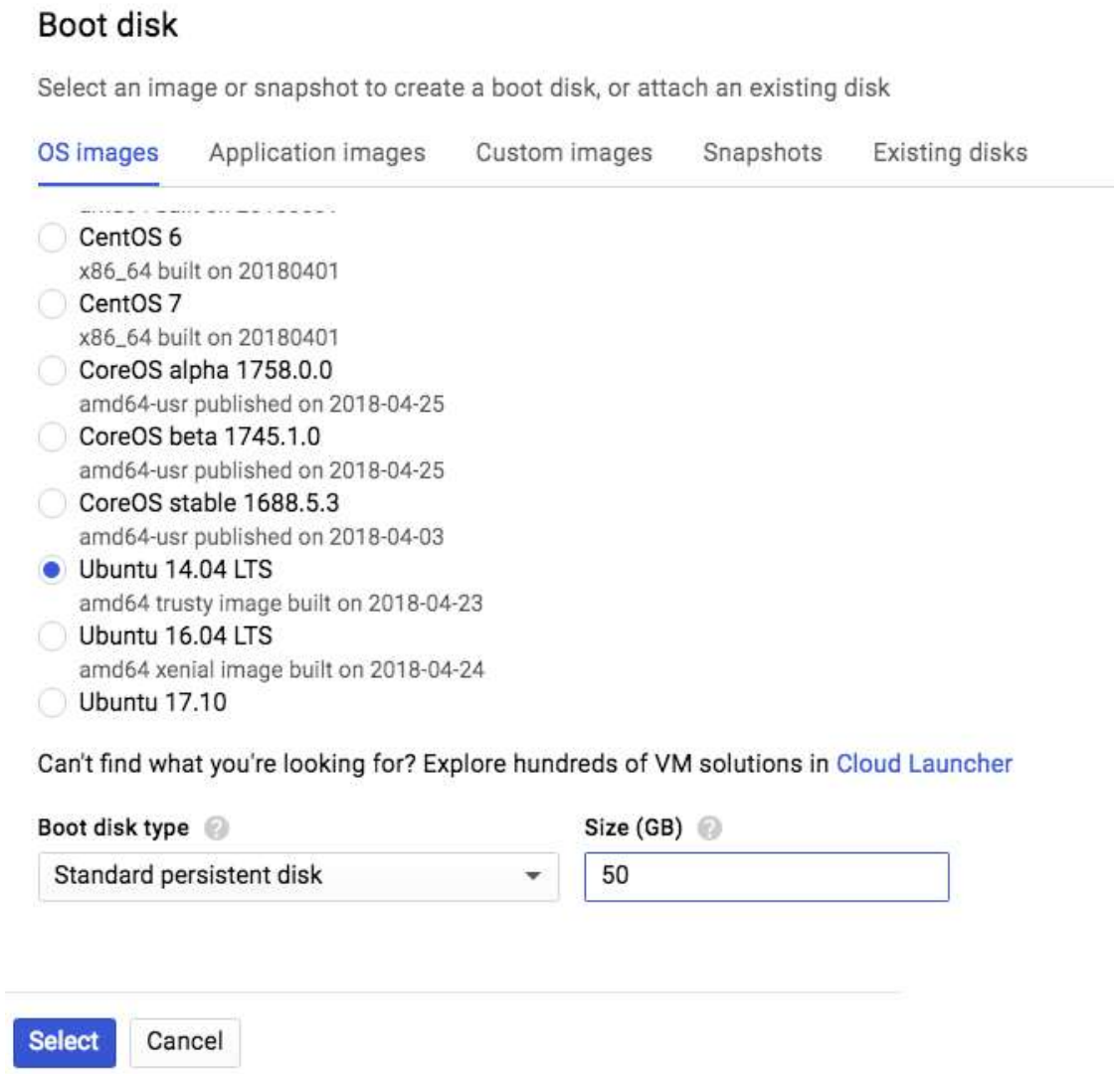


Create an instance with the following configuration



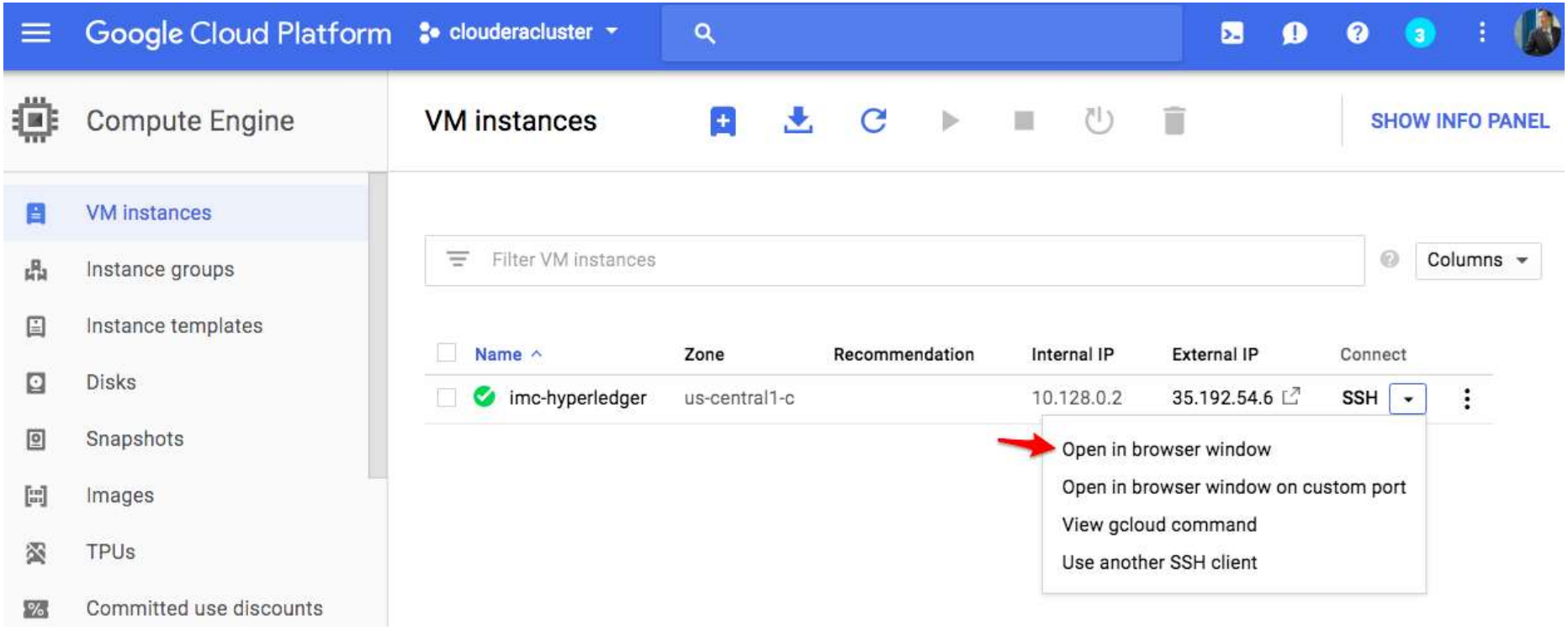


Select boot disk as Ubuntu 14.04 and 50 GB



Thanachart Numnonda, thanachart@imcinstitute.c

Connect via SSH in browser window



Connect to the instance

```
Connected, host fingerprint: ssh-rsa 2048 FC:E9:36:F1:43:FC:E6:D7:70:55:2E:3C:7F:C0:22:49:20:77:CD:8D
Welcome to Ubuntu 14.04.5 LTS (GNU/Linux 4.4.0-121-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

  System information as of Sat May  5 09:08:27 UTC 2018

  System load: 0.0               Memory usage: 1%   Processes:       66
  Usage of /:  11.4% of 9.81GB   Swap usage:   0%   Users logged in: 0

  Graph this data and manage this system at:
    https://landscape.canonical.com/

  Get cloud support with Ubuntu Advantage Cloud Guest:
    http://www.ubuntu.com/business/services/cloud

0 packages can be updated.
0 updates are security updates.

Your Hardware Enablement Stack (HWE) is supported until April 2019.


The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

thanachart@imc-hyperledger:~$
```


Thanachart Numnonda, thanachart@imcinstitute.c

Installing Hyperledger development environment

Thanachart Numnonda, thanachart@imcinstitute.c

Installing prerequisites

$ curl -O https://hyperledger.github.io/composer/latest/prereqs-ubu
$ chmod u+x prereqs-ubuntu.sh
$ ./prereqs-ubuntu.sh

```
thanachart@imc-hyperledger:~$ curl -O https://hyperledger.github.io/composer/latest/prereqs-ubuntu.sh
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  4151  100  4151    0     0  29972      0 --:--:-- --:--:-- --:--:-- 30079
thanachart@imc-hyperledger:~$ chmod u+x prereqs-ubuntu.sh
thanachart@imc-hyperledger:~$ ./prereqs-ubuntu.sh
Installing Hyperledger Composer prereqs for Ubuntu trusty
# Updating package lists
gpg: keyring `/tmp/tmp8mglxygq/secring.gpg' created
gpg: keyring `/tmp/tmp8mglxygq/pubring.gpg' created
gpg: requesting key E1DF1F24 from hkp server keyserver.ubuntu.com
gpg: /tmp/tmp8mglxygq/trustdb.gpg: trustdb created
gpg: key E1DF1F24: public key "Launchpad PPA for Ubuntu Git Maintainers" imported
gpg: Total number processed: 1
```

Installing the Hyperledger Composer tools

$ npm install -g composer-cli
$ npm install -g composer-rest-server
$ npm install -g generator-hyperledger-composer
$ npm install -g yo
$ npm install -g composer-playground

```
> grpc@1.10.1 install /home/thanachart/.nvm/versions/node/v8.11.1/lib/node_modules/composer-playground/node_modules/
pc
> node-pre-gyp install --fallback-to-build --library=static_library

[grpc] Success: "/home/thanachart/.nvm/versions/node/v8.11.1/lib/node_modules/composer-playground/node_modules/grpc/
c/node/extension_binary/node-v57-linux-x64-glibc/grpc_node.node" is installed via remote
npm WARN composer-wallet-inmemory@0.19.4 requires a peer of composer-common@0.19.3 but none is installed. You must i
tall peer dependencies yourself.
npm WARN composer-wallet-filesystem@0.19.4 requires a peer of composer-common@0.19.3 but none is installed. You must
nstall peer dependencies yourself.

+ composer-playground@0.19.4
added 599 packages from 473 contributors in 34.224s
thanachart@imc-hyperledger:~$
```

Installing Hyperledger Fabric

$ mkdir ~/fabric-dev-servers && cd ~/fabric-dev-servers
$ curl -O https://raw.githubusercontent.com/hyperledger/composer-tools/master/packages/fabric-dev-servers/fabric-dev-servers.tar.gz
$ tar -xvf fabric-dev-servers.tar.gz
$ ./downloadFabric.sh

```
thanachart@imc-hyperledger:~$ mkdir ~/fabric-dev-servers && cd ~/fabric-dev-servers
thanachart@imc-hyperledger:~/fabric-dev-servers$ curl -O https://raw.githubusercontent.com/hyperledger/composer-tools/
master/packages/fabric-dev-servers/fabric-dev-servers.tar.gz
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 23416  100 23416    0     0   128k      0 --:--:-- --:--:-- --:--:--  129k
thanachart@imc-hyperledger:~/fabric-dev-servers$ tar -xvf fabric-dev-servers.tar.gz
package.json
_loader.sh
createComposerProfile.sh
createPeerAdminCard.sh
downloadFabric.sh
startFabric.sh
stopFabric.sh
```

Starting Hyperledger-Fabric
&
Hyperledger composer playground

Thanachart Numnonda, thanachart@imcinstitute.c

Starting Hyperledger-Fabric

$ ./startFabric.sh
$ ./createPeerAdminCard.sh

```
thanachart@imc-hyperledger:~/fabric-dev-servers$ ./createPeerAdminCard.sh
Development only script for Hyperledger Fabric control
Running 'createPeerAdminCard.sh'
FABRIC_VERSION is unset, assuming hlfv11
FABRIC_START_TIMEOUT is unset, assuming 15 (seconds)

Using composer-cli at v0.19.4

Successfully created business network card file to
        Output file: /tmp/PeerAdmin@hlfv1.card

Command succeeded


Successfully imported business network card
        Card file: /tmp/PeerAdmin@hlfv1.card
        Card name: PeerAdmin@hlfv1

Command succeeded

The following Business Network Cards are available:

Connection Profile: hlfv1
```

Card Name	UserId	Business Network
PeerAdmin@hlfv1	PeerAdmin	

Viewing running process

$ docker ps -a

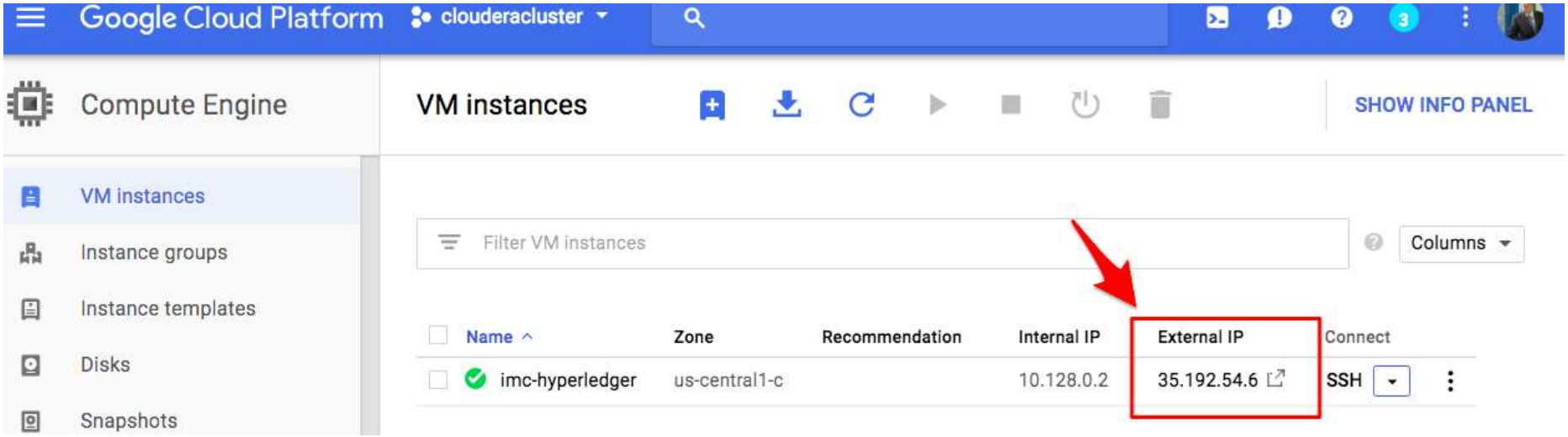
```
thanachart@imc-hyperledger:~/fabric-dev-servers$ docker ps -a
CONTAINER ID        IMAGE                                    COMMAND                  CREATED              STATUS
          PORTS                                            NAMES
43afe08cebff        hyperledger/fabric-peer:x86_64-1.1.0      "peer node start"        About a minute ago   Up About a
 minute   0.0.0.0:7051->7051/tcp, 0.0.0.0:7053->7053/tcp   peer0.org1.example.com
fa066327c38a        hyperledger/fabric-couchdb:x86_64-0.4.6   "tini -- /docker-ent…"   About a minute ago   Up About a
 minute   4369/tcp, 9100/tcp, 0.0.0.0:5984->5984/tcp       couchdb
d3914b516484        hyperledger/fabric-orderer:x86_64-1.1.0   "orderer"                About a minute ago   Up About a
 minute   0.0.0.0:7050->7050/tcp                           orderer.example.com
a321e0e313bc        hyperledger/fabric-ca:x86_64-1.1.0        "sh -c 'fabric-ca-se…"   About a minute ago   Up About a
 minute   0.0.0.0:7054->7054/tcp                           ca.org1.example.com
```

Starting composer-playground

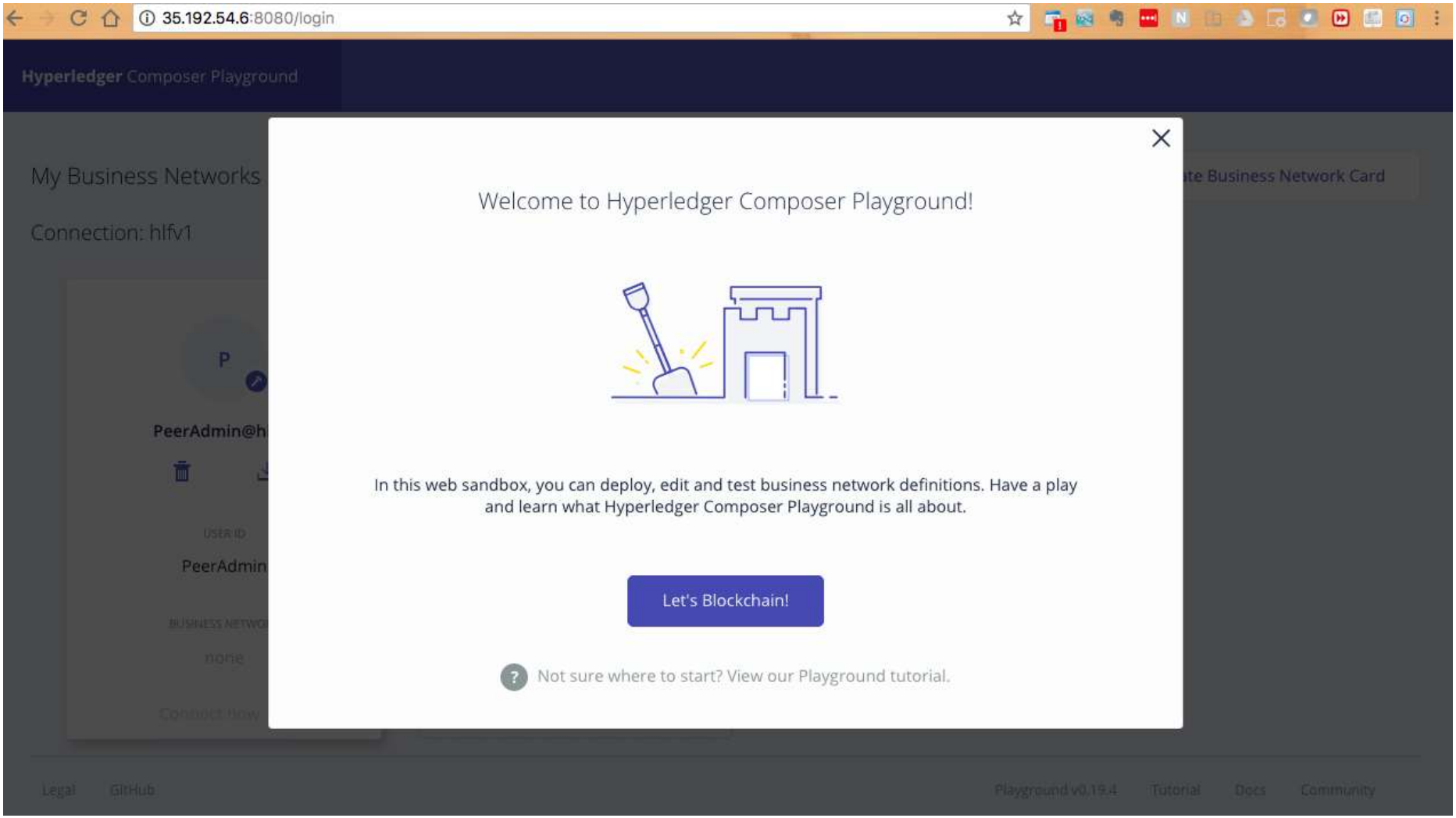
$ composer-playground

```
thanachart@imc-hyperledger:~/fabric-dev-servers$ composer-playground
info: [Hyperledger-Composer] :LoadModule                :loadModule()              Loading composer-wallet-filesystem f
rom /home/thanachart/.nvm/versions/node/v8.11.1/lib/node_modules/composer-playground/node_modules/composer-wallet-file
system
info: [Hyperledger-Composer] :PlaygroundAPI             :createServer()            Playground API started on port 8080
```

Obtain a server's external IP



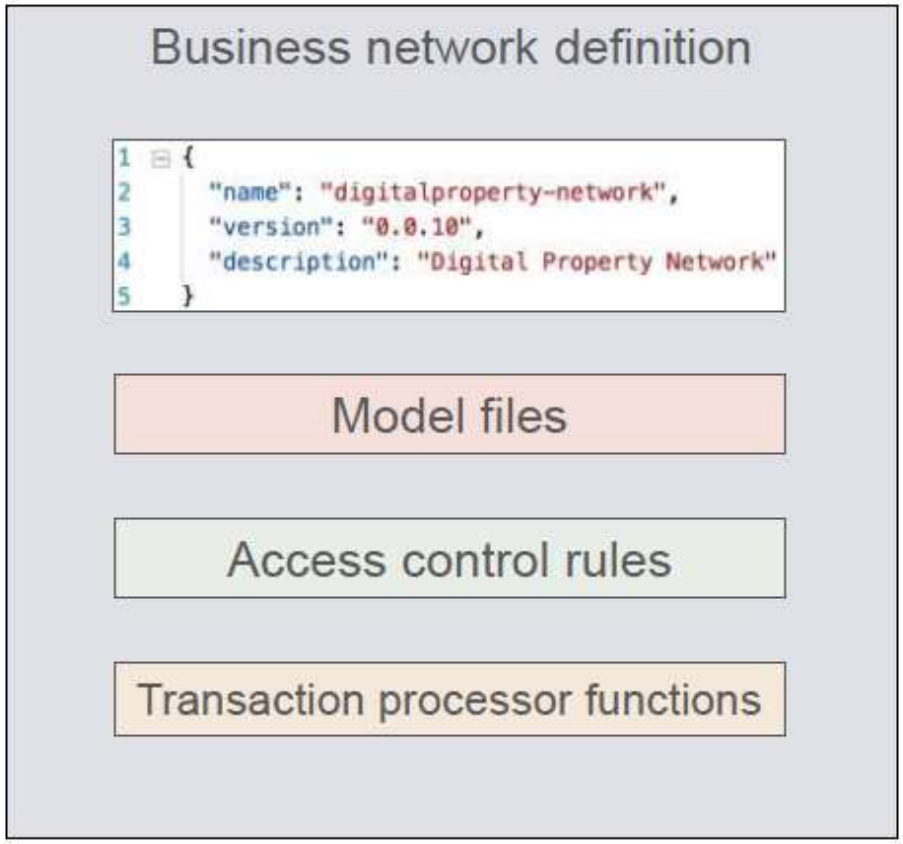
Launch web-playground (http://ip-address:8080)



Running an example business network

Thanachart Numnonda, thanachart@imcinstitute.c

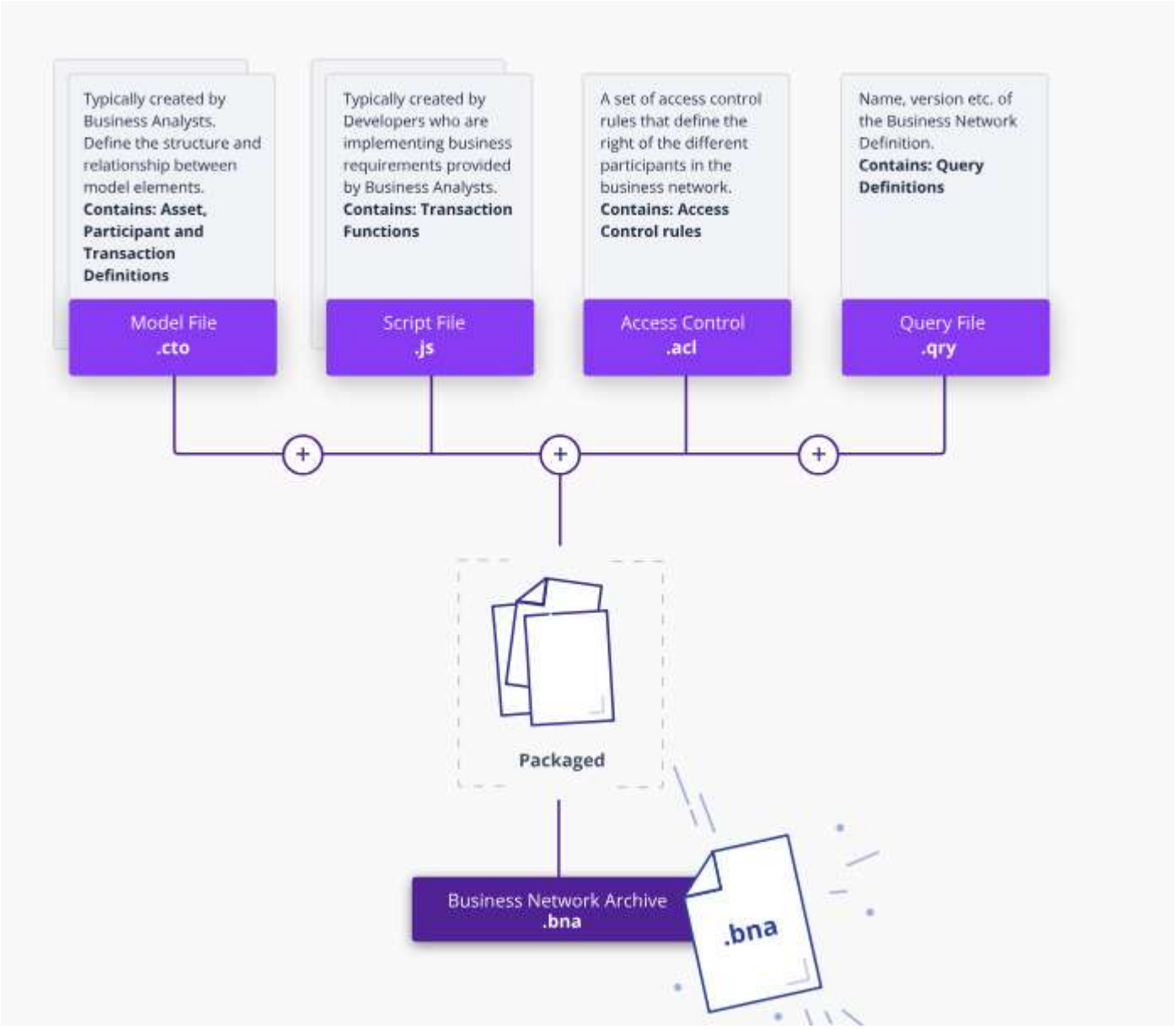
Business network definition



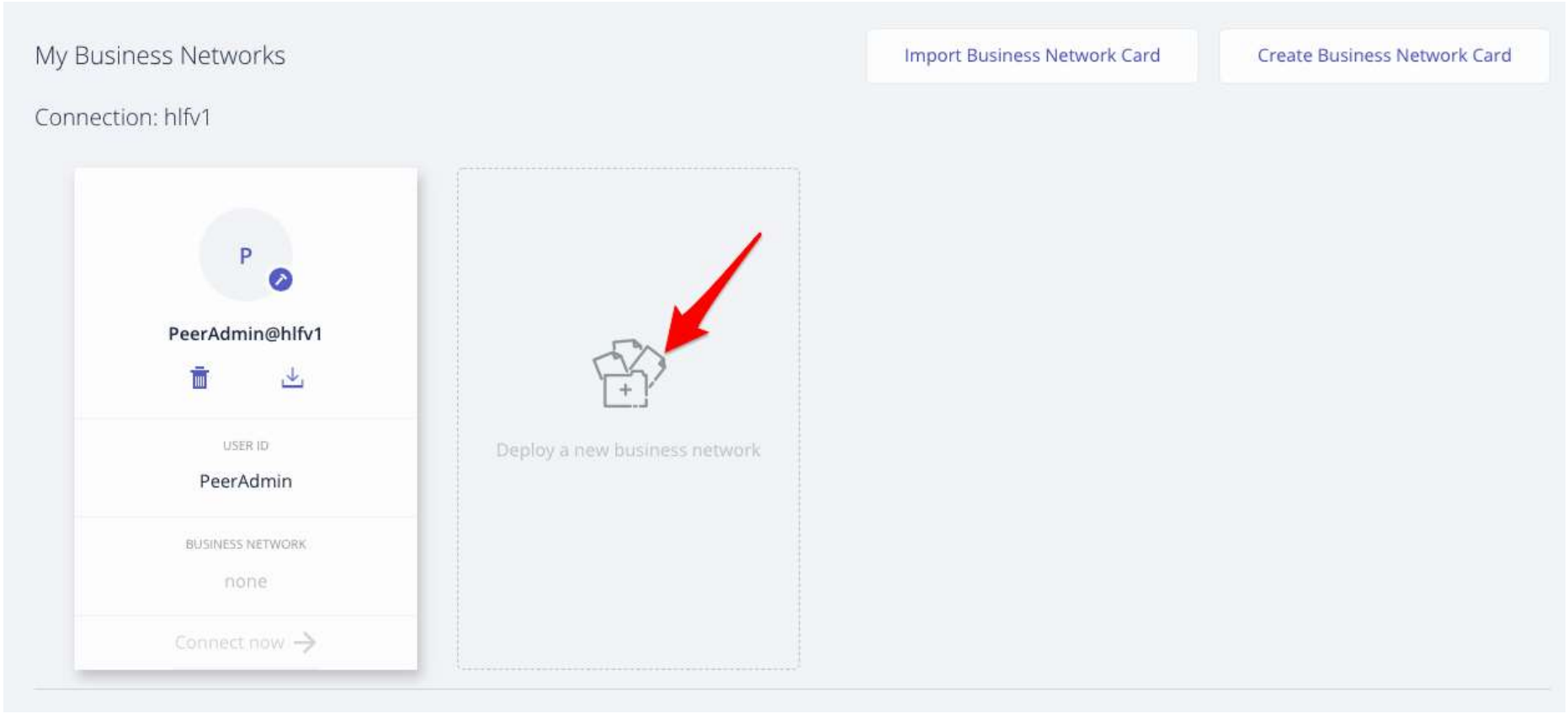
Each business network definition has metadata associated with it – at least a name and version.

The metadata is stored in a package.json file. Each business network definition is a npm module.

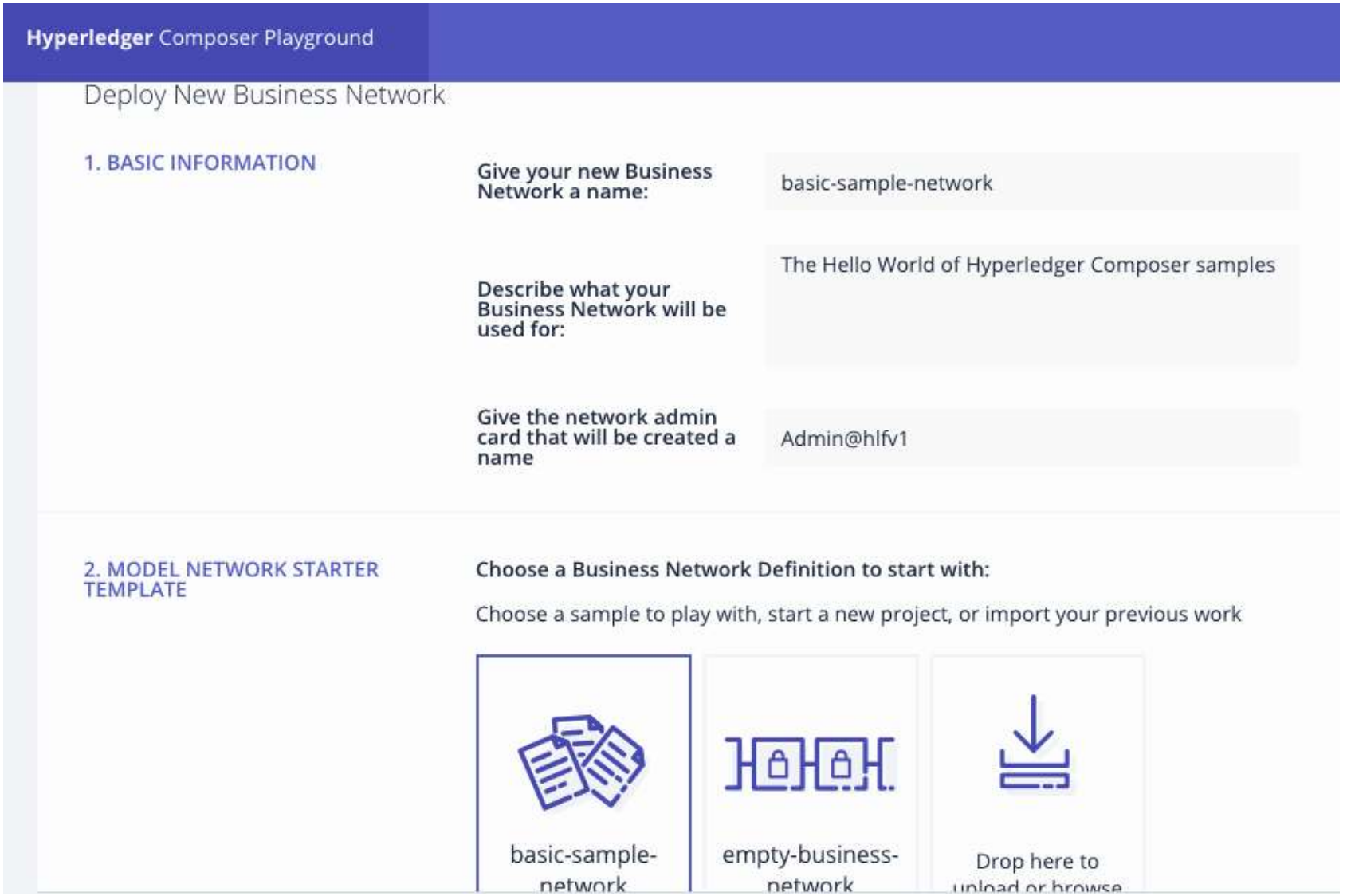
The business network definition contains all of the model files, access control rules, and transaction processor functions for a business network.


Typically created by Business Analysts. Define the structure and relationship between model elements. **Contains: Asset, Participant and Transaction Definitions**
Model File .cto
Typically created by Developers who are implementing business requirements provided by Business Analysts. **Contains: Transaction Functions**
Script File .js
A set of access control rules that define the right of the different participants in the business network. **Contains: Access Control rules**
Access Control .acl
Name, version etc. of the Business Network Definition. **Contains: Query Definitions**
Query File .qry
Packaged
Business Network Archive .bna
.bna

Click Deploy a new business network

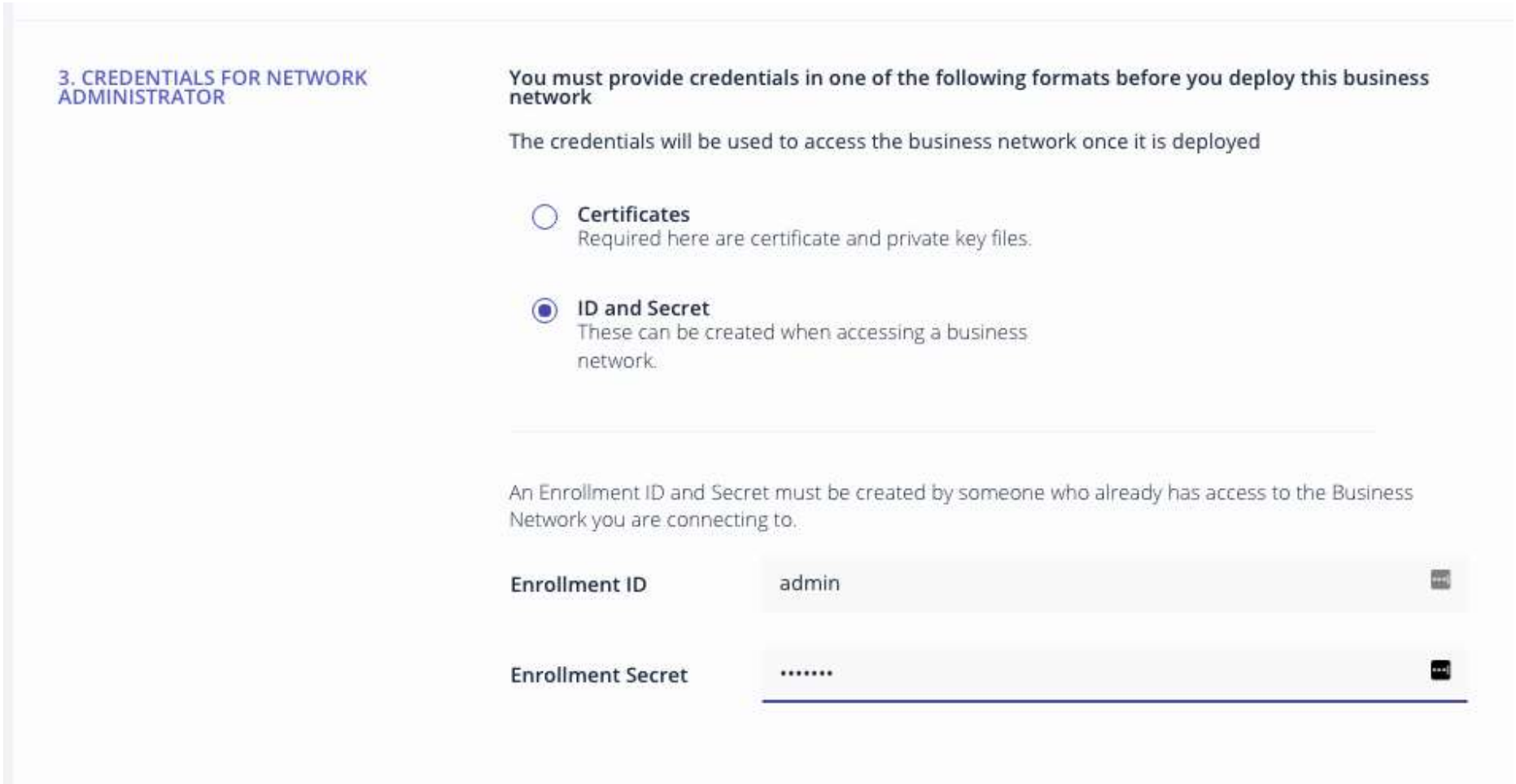


Provide basic information and select basic-sample -network (provide the network admin as Admin@hlfv1)

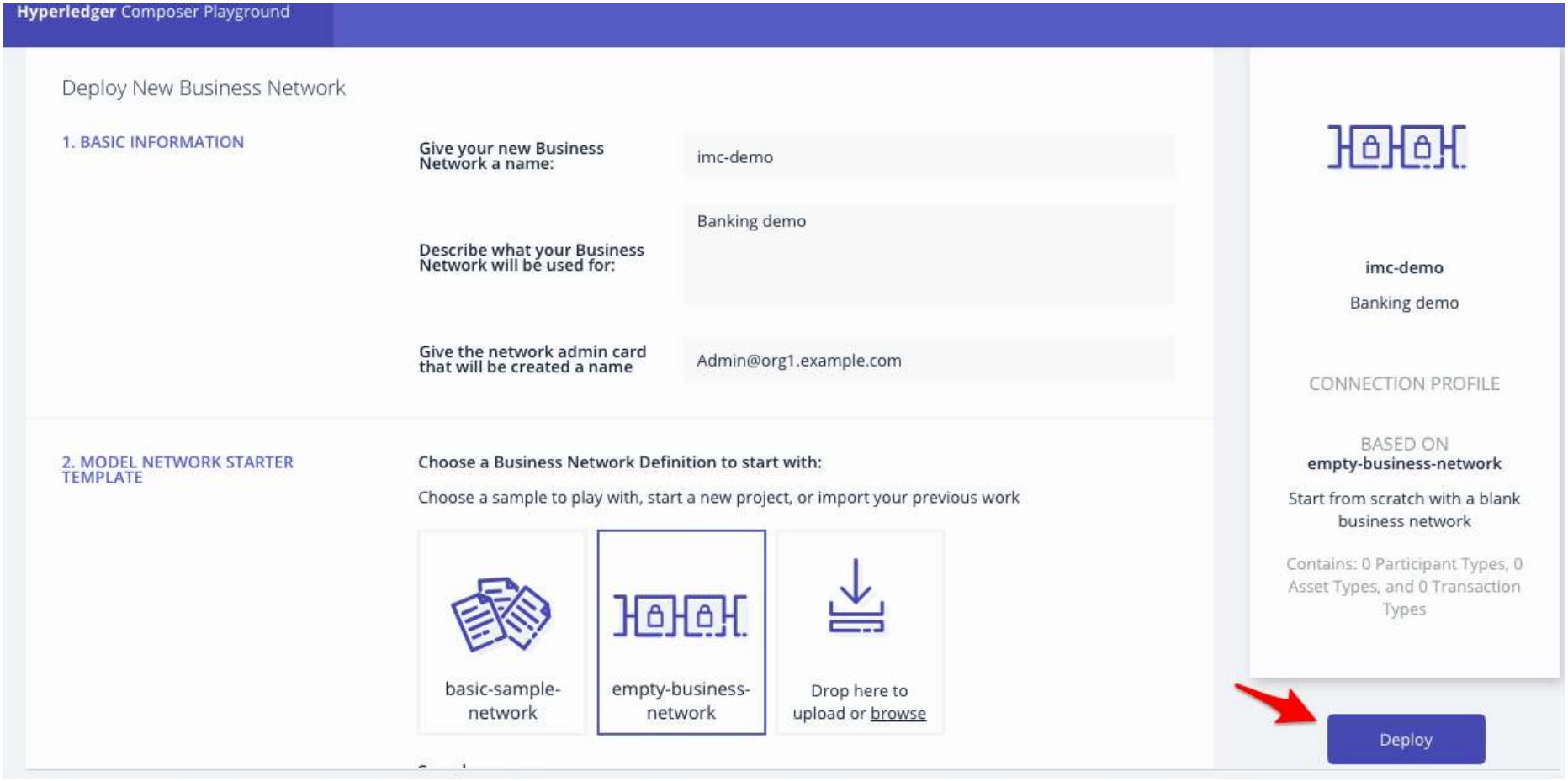


Provide credentials information

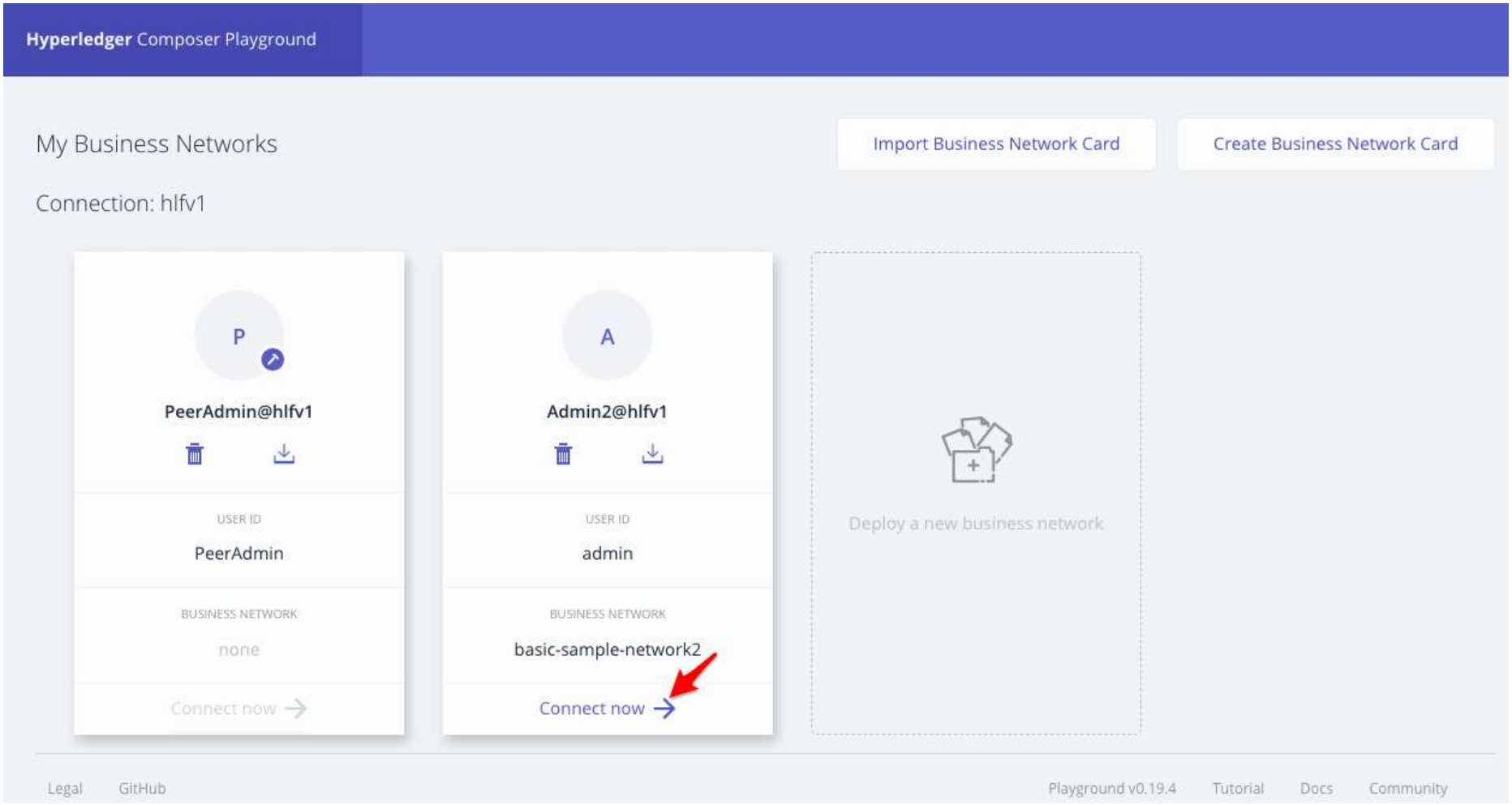
(Assign the Enrolment id: admin Enrolment secret : adminpw)



Now the network is defined, click Deploy.



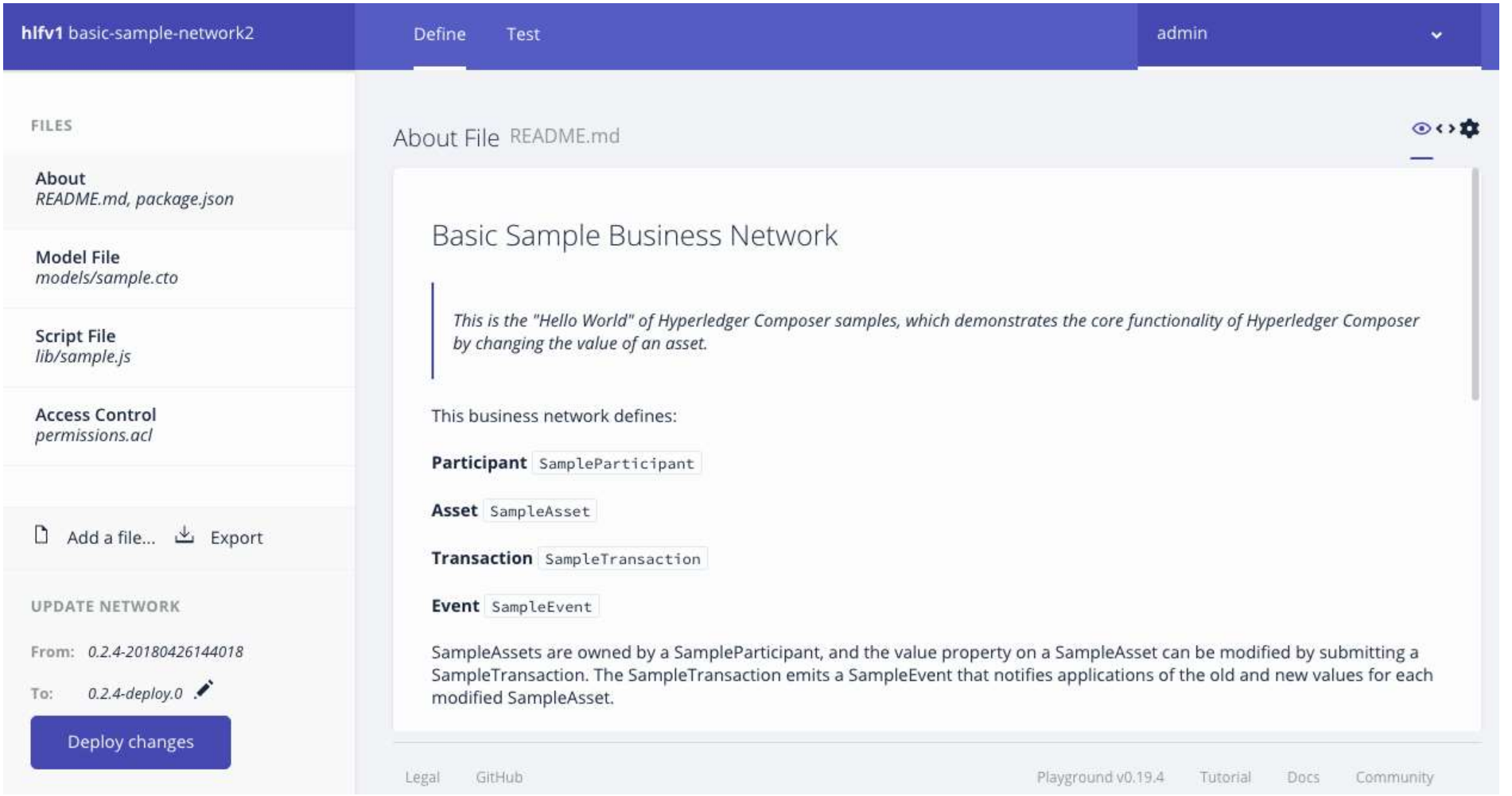
The new business network will be shown, click Connect now



Open another terminal console, Type command docker ps -a, You will see another container is running

```
thanachart@imchyperledger:~$ docker ps -a
CONTAINER ID        IMAGE
                                    COMMAND                  CREATED             STATUS              PORTS
                                  NAMES
0f8cb042c530        dev-peer0.org1.example.com-basic-sample-network-0.2.4-20180426144018-f192bdbf3887f510d7498828d7e20
23ad9494488b37e8c98465841b0e2982e06   "/bin/sh -c 'cd /usr…"   About a minute ago   Up About a minute
                                  dev-peer0.org1.example.com-basic-sample-network-0.2.4-20180426144018
9ac9c739068d        hyperledger/fabric-peer:x86_64-1.1.0
                                    "peer node start"        18 minutes ago      Up 18 minutes       0.0.0.0:7051->
7051/tcp, 0.0.0.0:7053->7053/tcp   peer0.org1.example.com
610fdf23f222        hyperledger/fabric-couchdb:x86_64-0.4.6
                                    "tini -- /docker-ent…"   18 minutes ago      Up 18 minutes       4369/tcp, 9100
/tcp, 0.0.0.0:5984->5984/tcp       couchdb
f61e54c98552        hyperledger/fabric-ca:x86_64-1.1.0
                                    "sh -c 'fabric-ca-se…"   18 minutes ago      Up 18 minutes       0.0.0.0:7054->
7054/tcp                           ca.org1.example.com
e681d774d6c1        hyperledger/fabric-orderer:x86_64-1.1.0
                                    "orderer"                18 minutes ago      Up 18 minutes       0.0.0.0:7050->
```


thanachart@imcinstitute.com

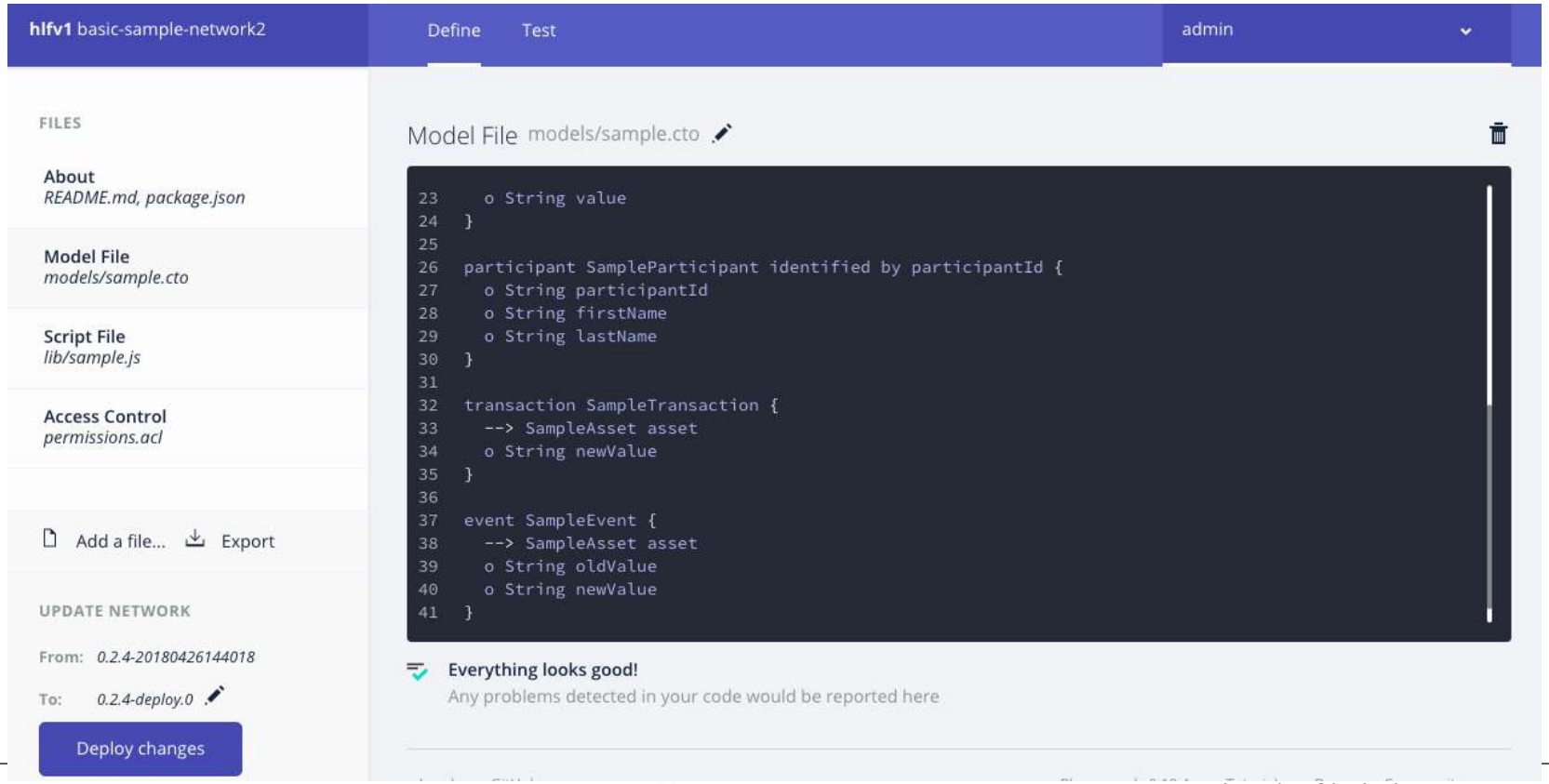


thanachart@imcinstitute.com

Exploring a business network

Viewing the following files:

- Model
- Script
- Access control



thanachart@imcinstitute.com

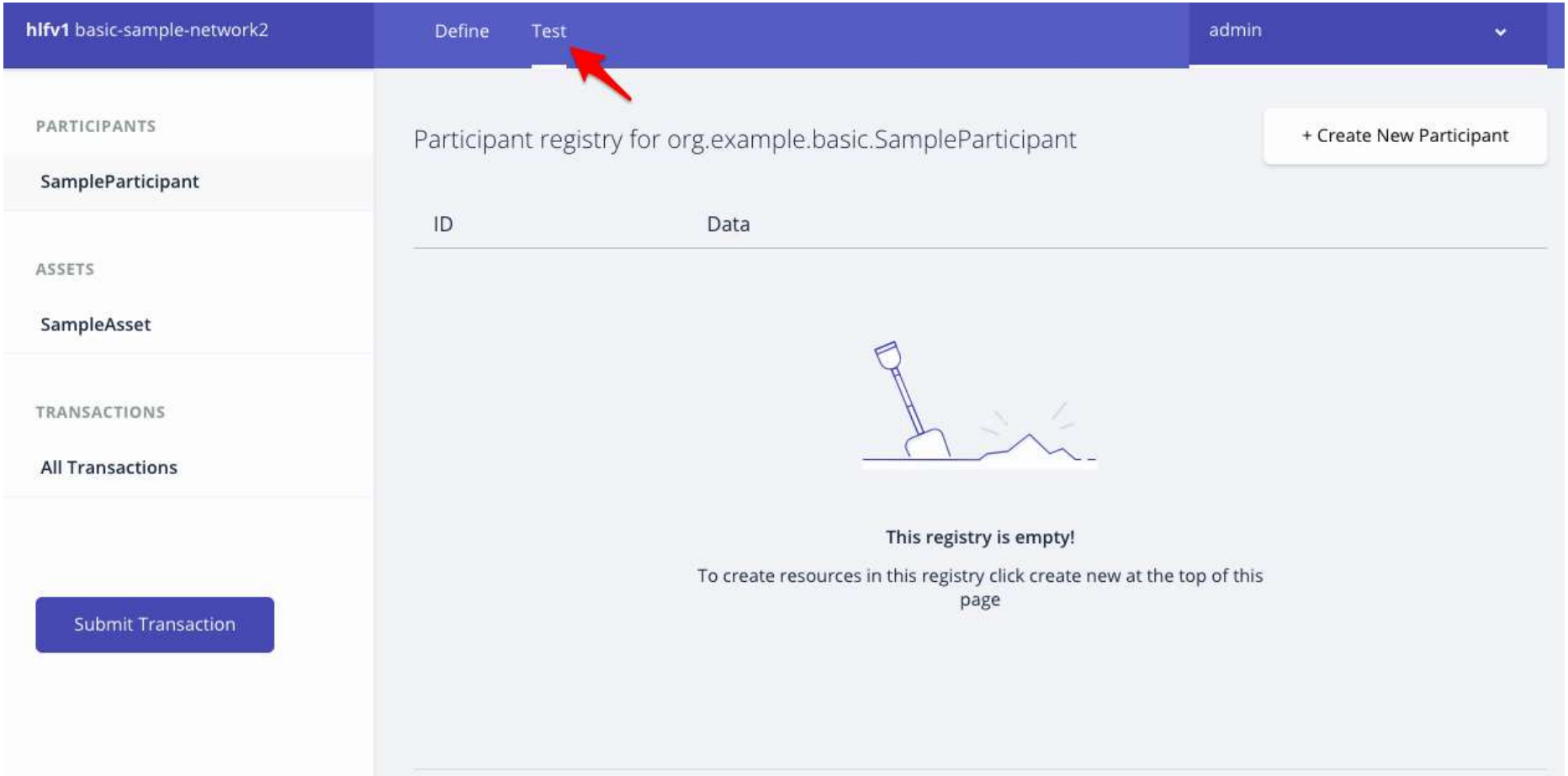
Basic-sample-network

Definition

- Asset
  - SampleAsset
- Participant
  - SampleParticipant
- Transaction
  - SampleTransaction
- Event
  - SampleEvent

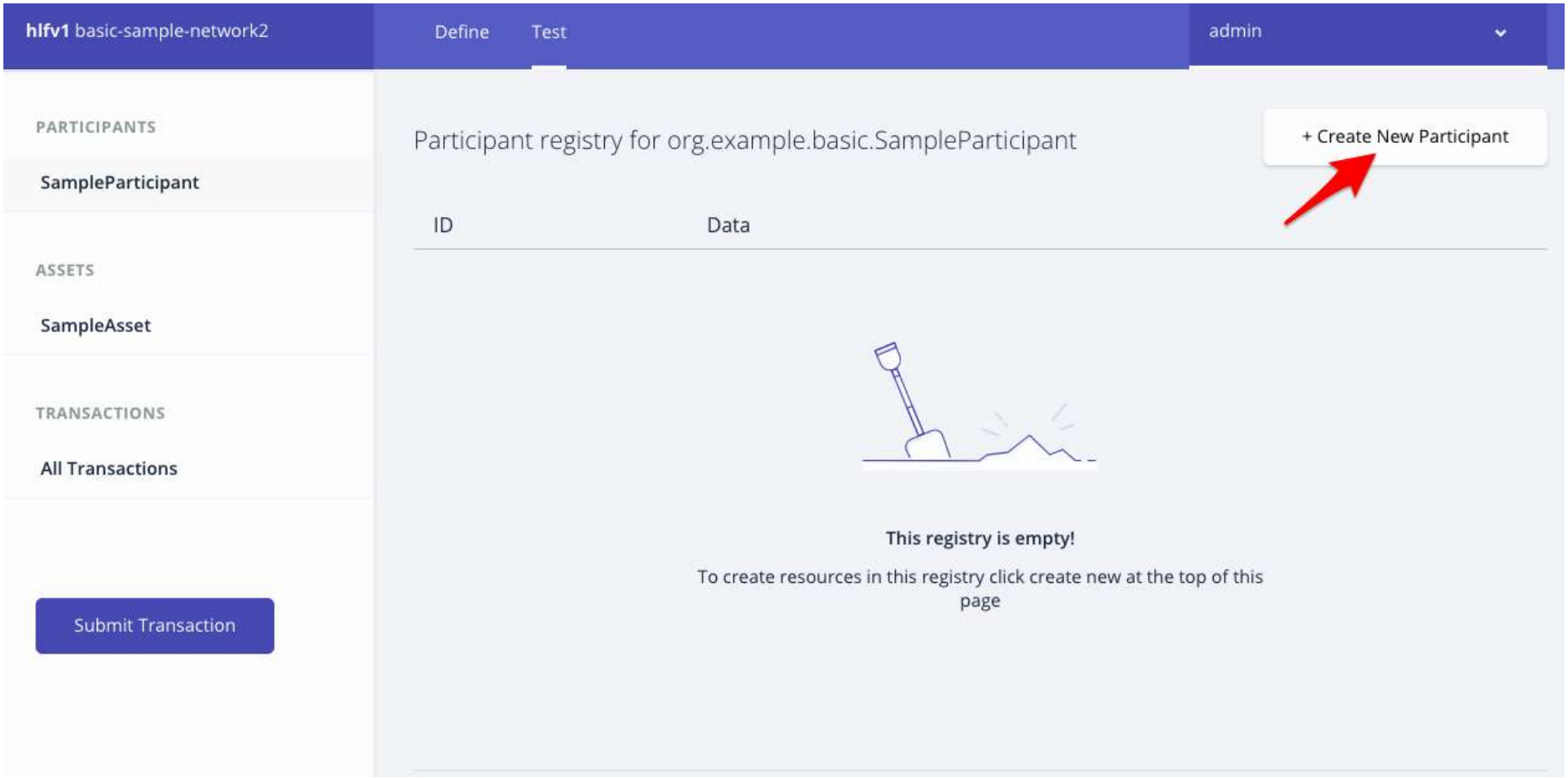
thanachart@imcinstitute.com

Testing the business network definition, Click Test tab



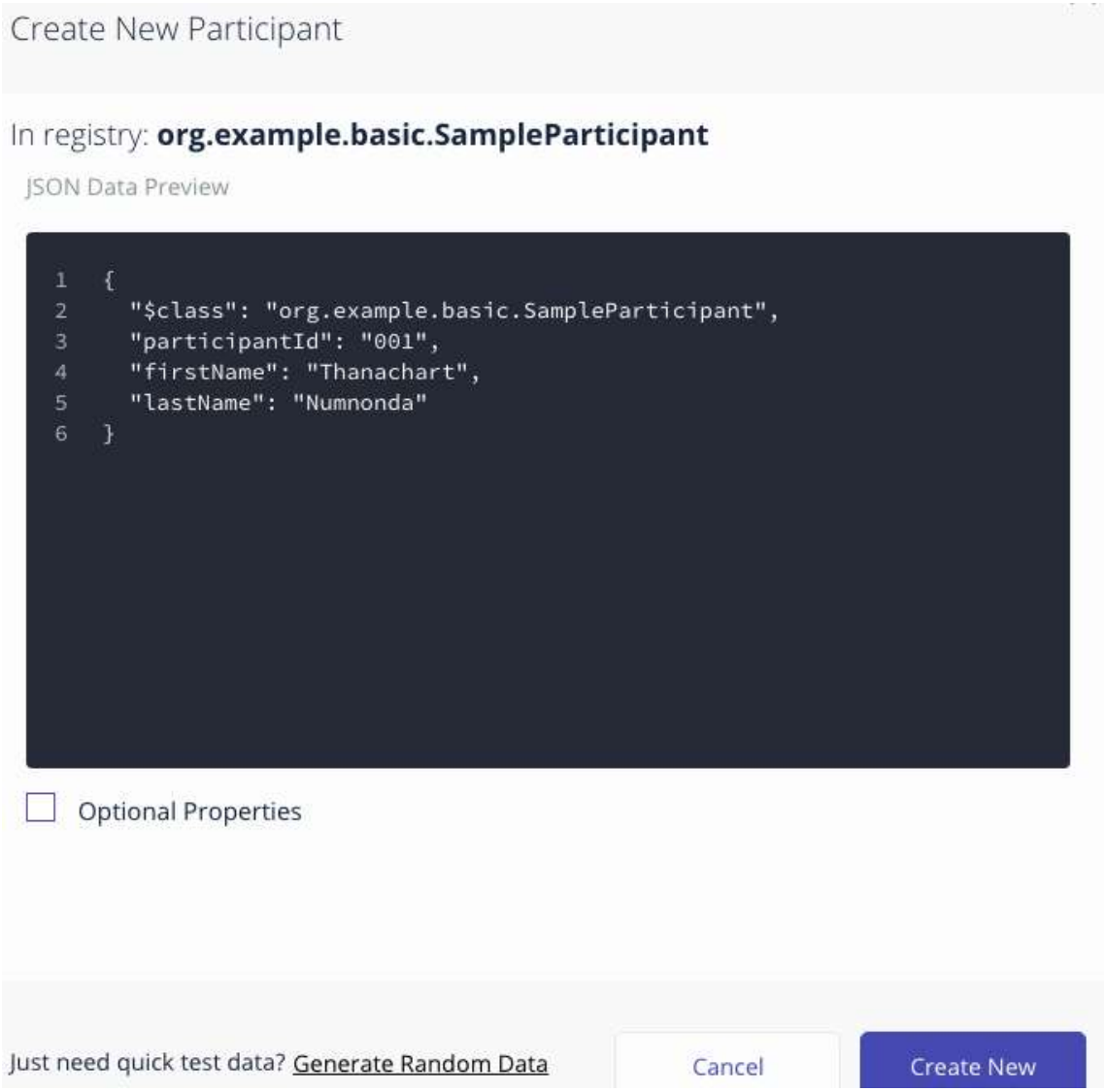
thanachart@imcinstitute.com

Click Create New Participant



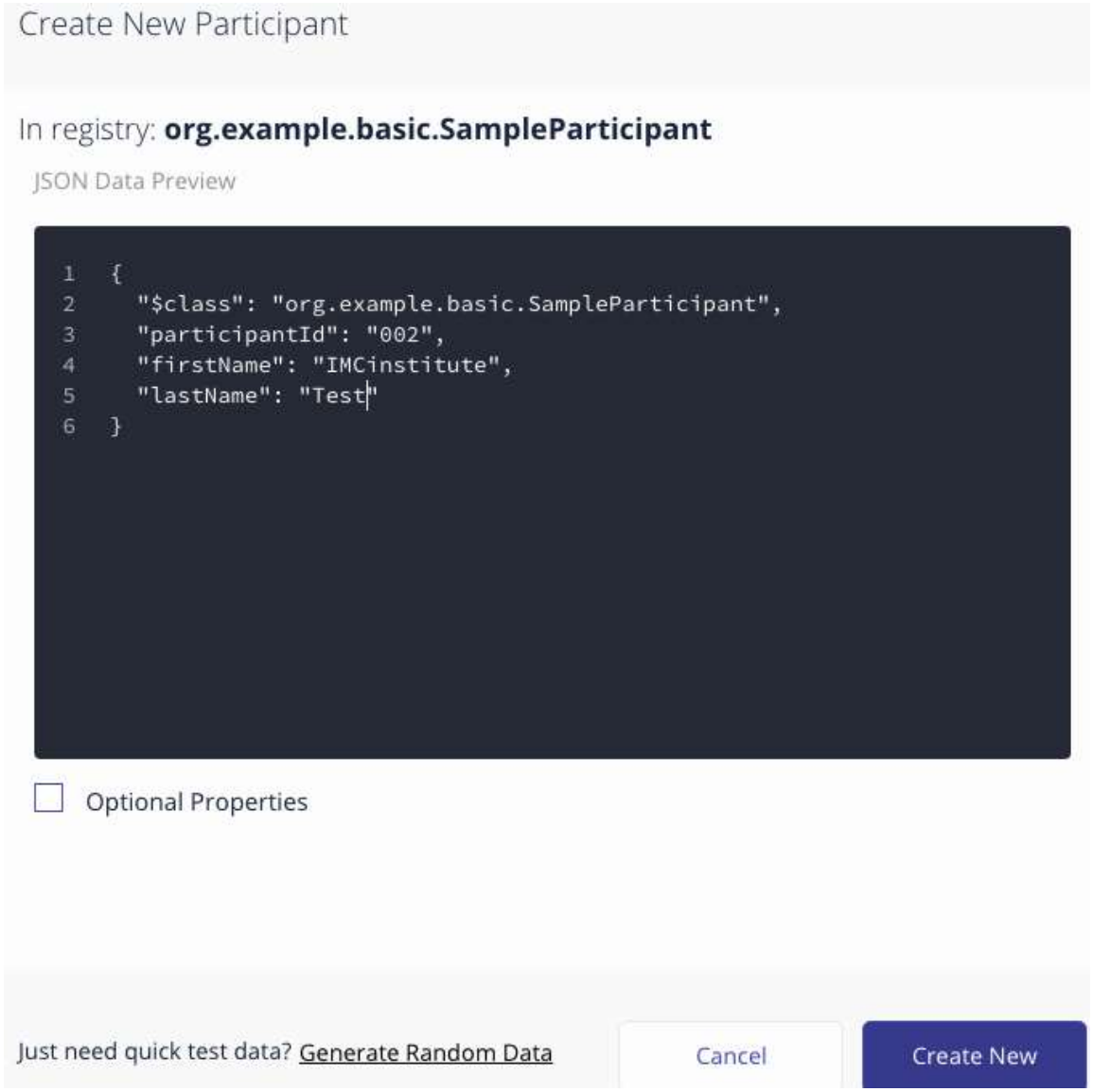
thanachart@imcinstitute.com

Enter the first participant, the click Create New



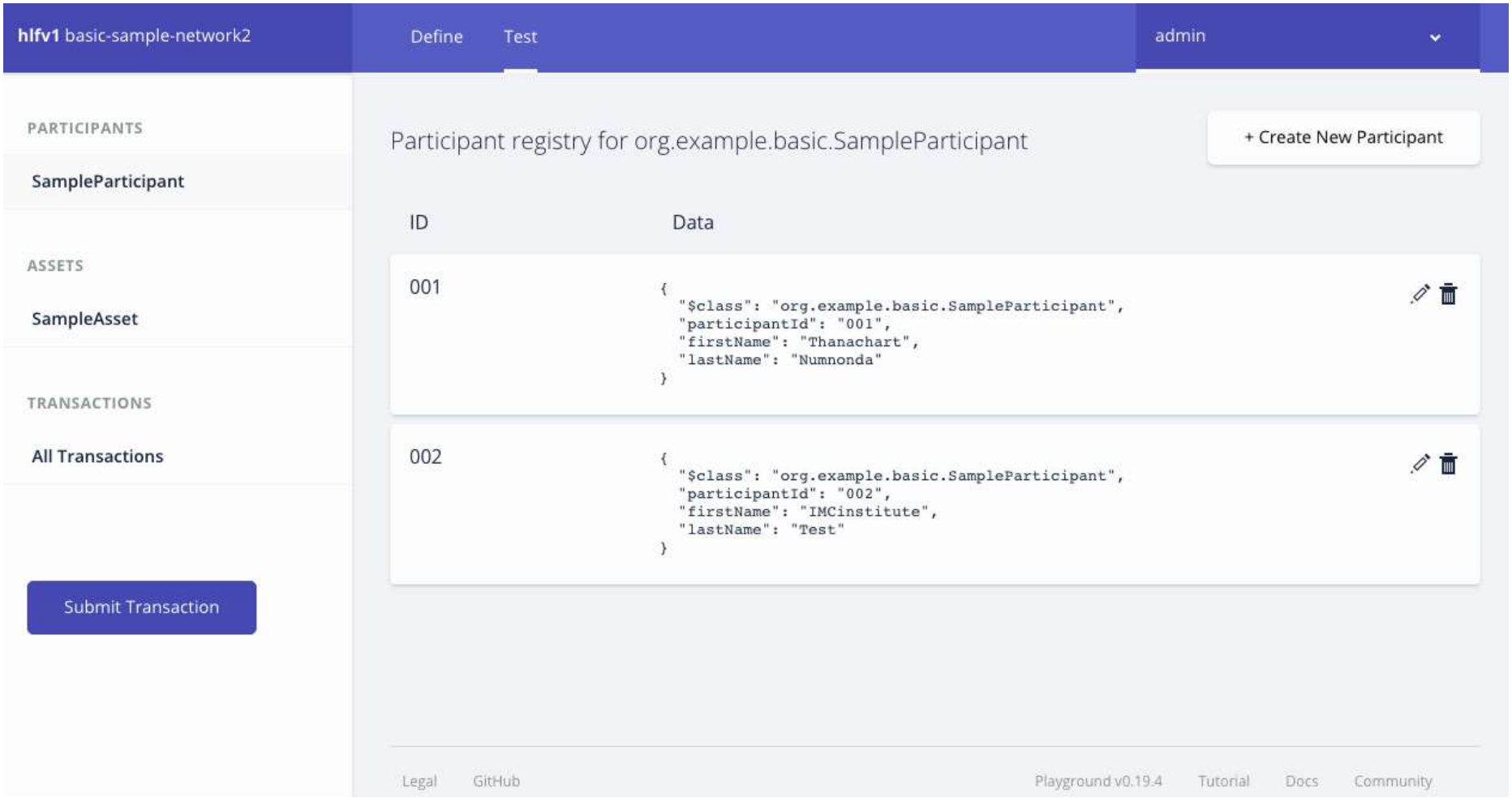
thanachart@imcinstitute.com

Enter the second participant, the click Create New



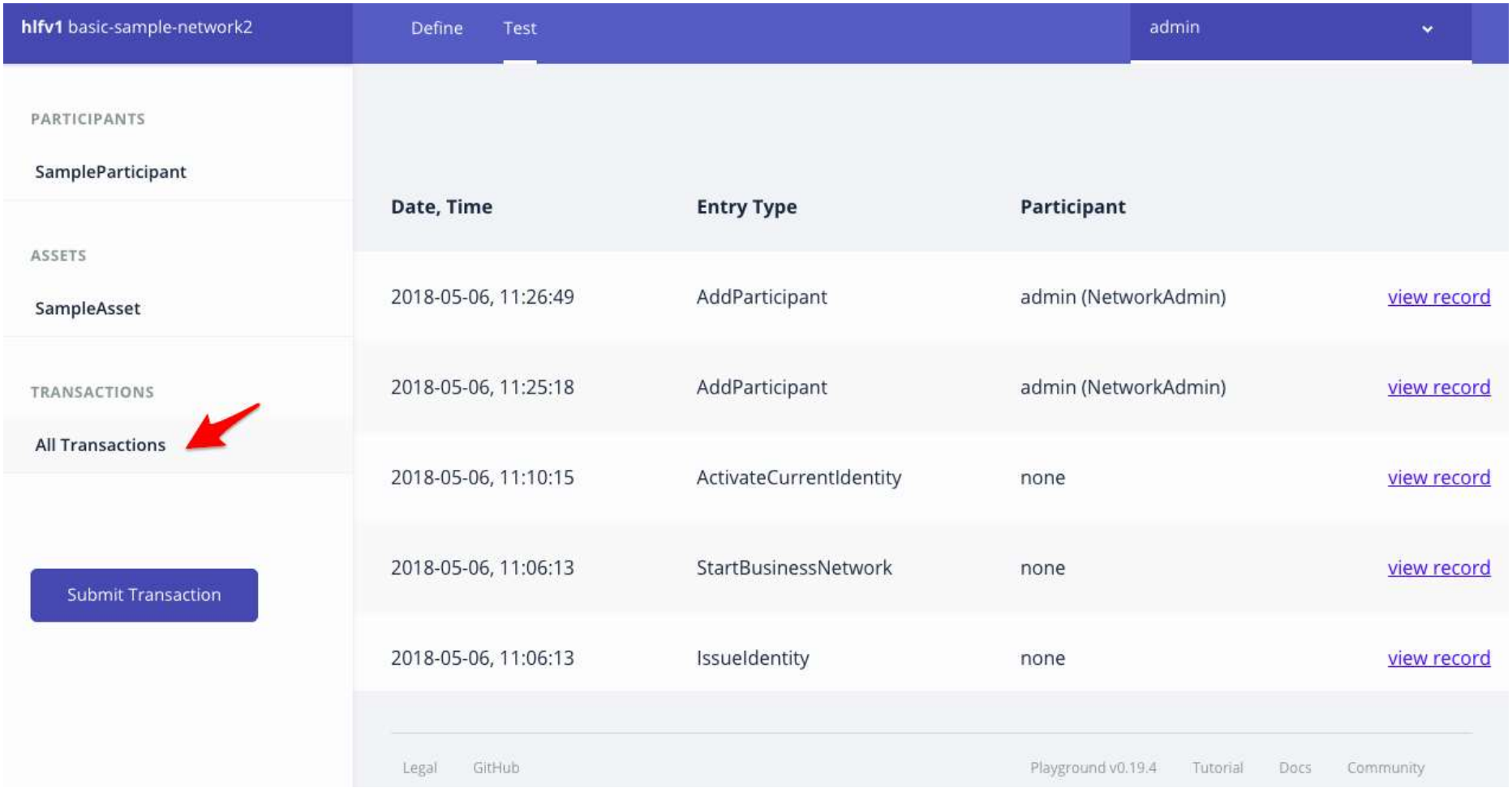
thanachart@imcinstitute.com

A list of participants will be shown



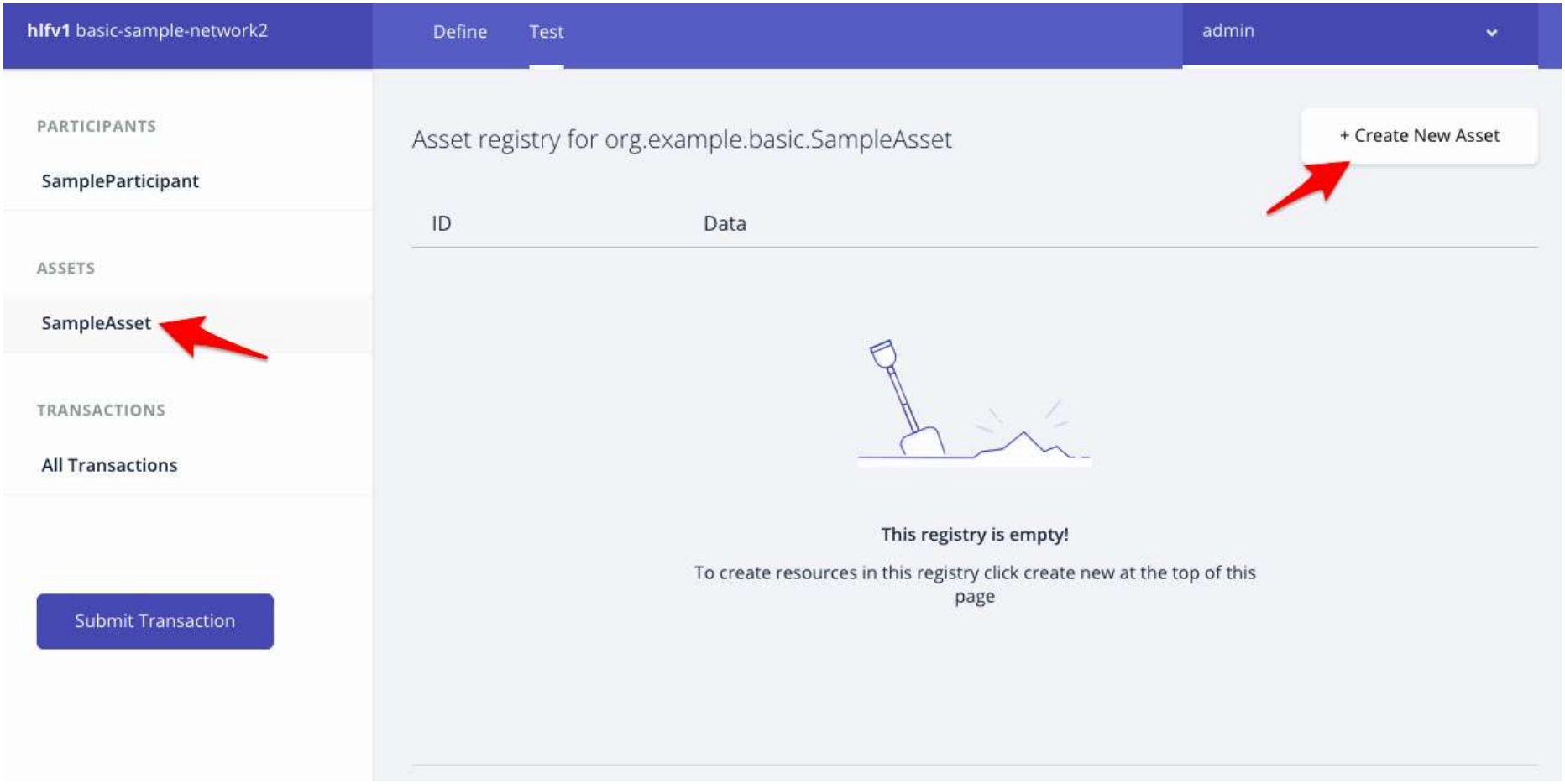
thanachart@imcinstitute.com

All transactions show events



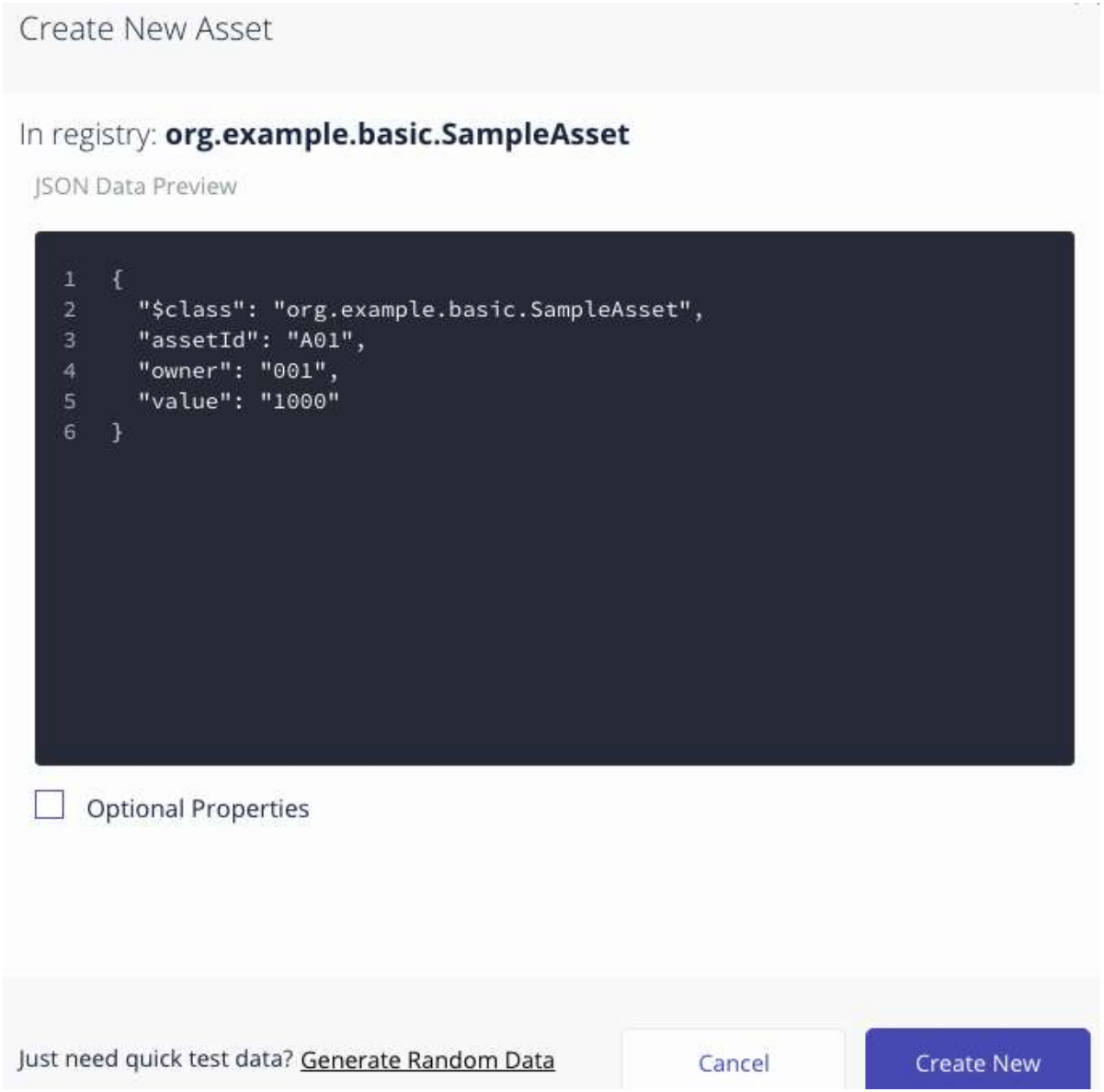
thanachart@imcinstitute.com

Select SampleAsset, the click Create New Asset



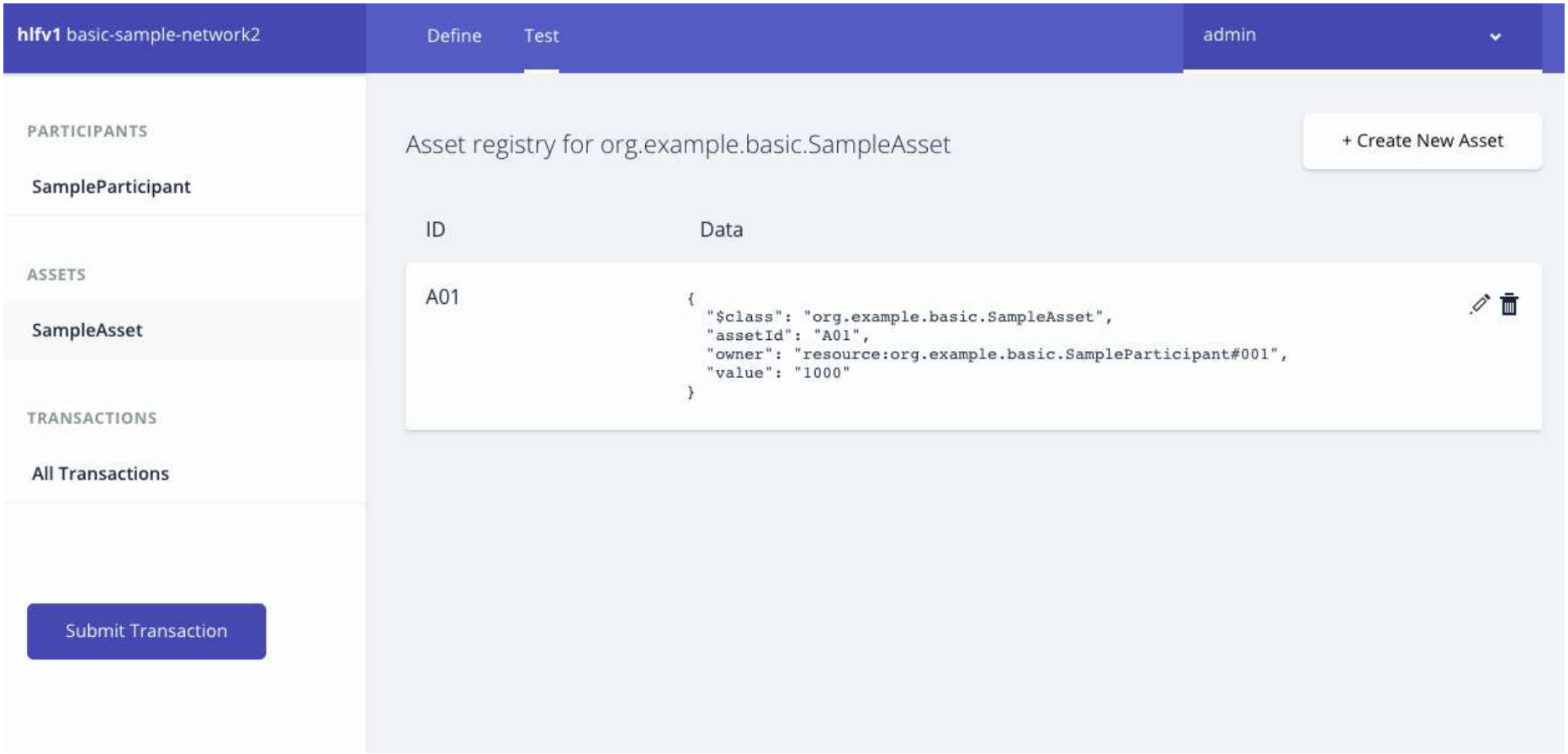
thanachart@imcinstitute.com

Enter the new asset, the click Create New



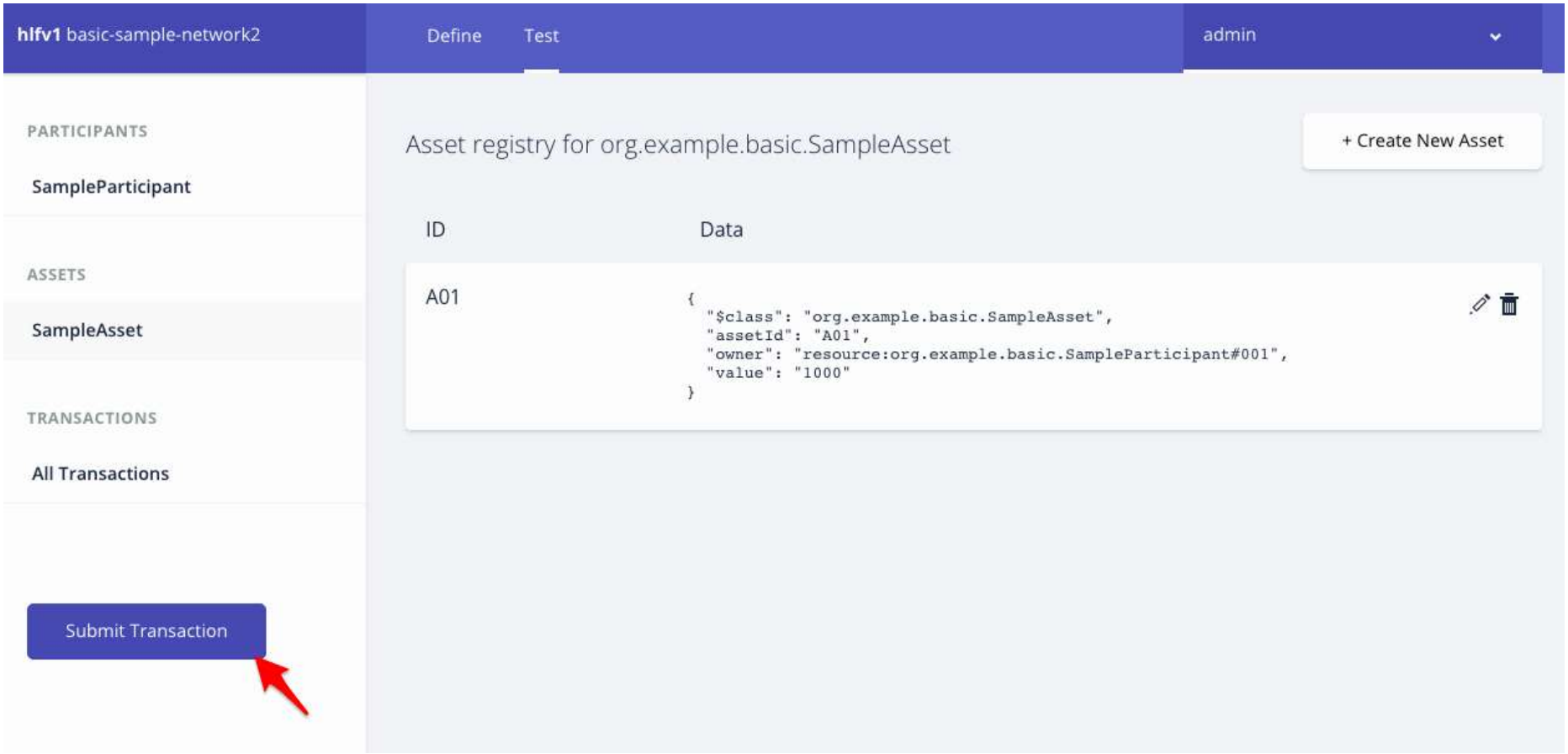
thanachart@imcinstitute.com

A list of assets will be shown



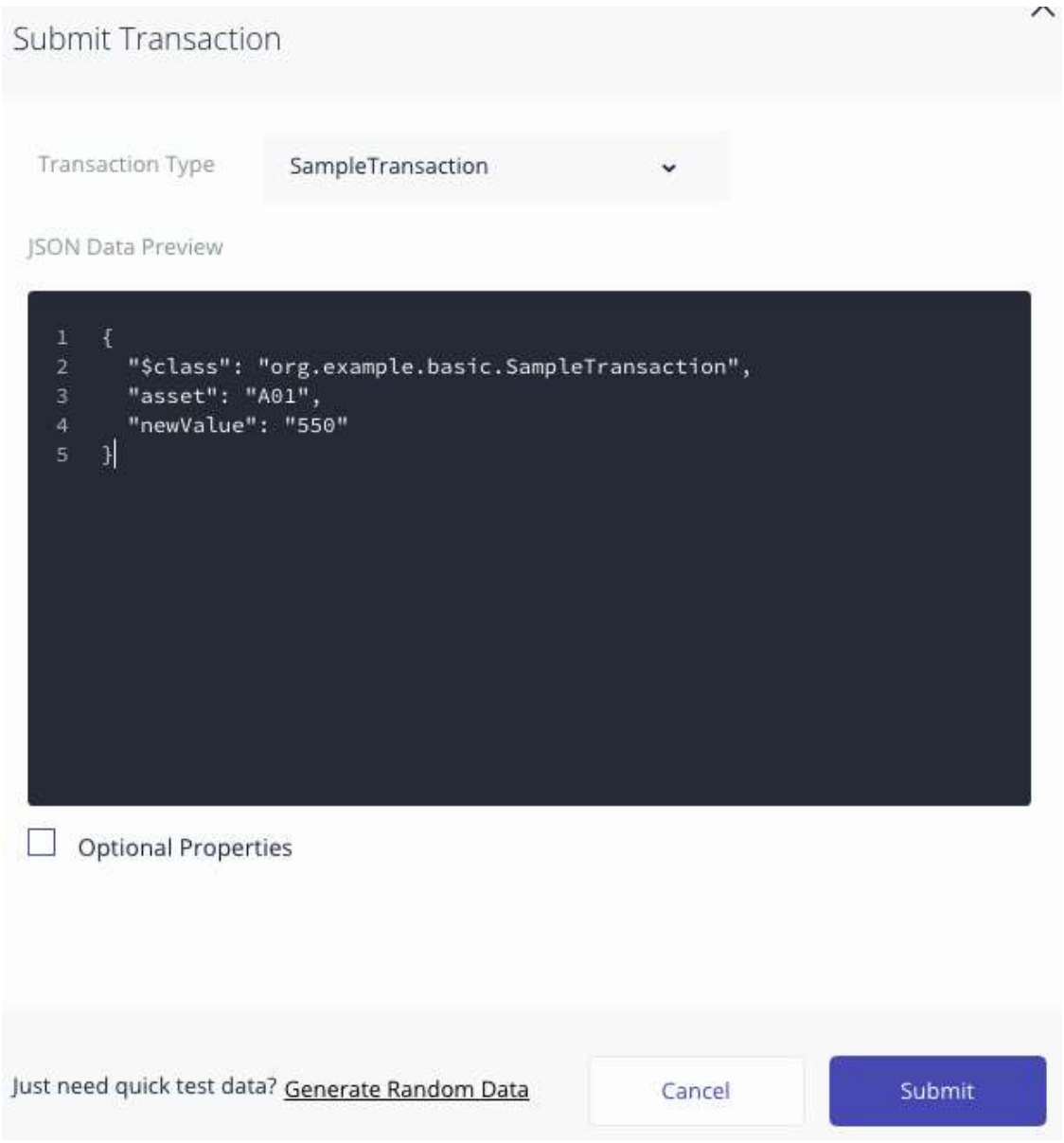
thanachart@imcinstitute.com

Click Submit Transaction



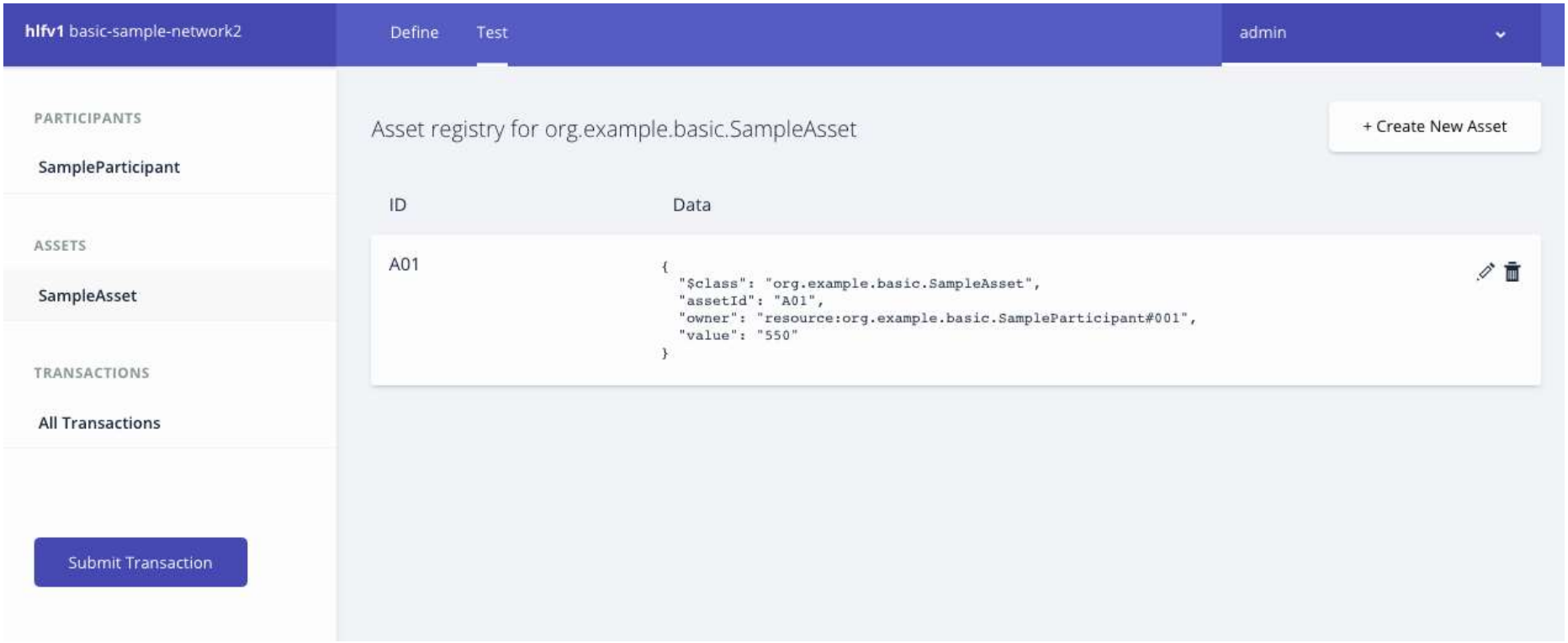
thanachart@imcinstitute.com

Enter the transaction information, the click Submit



thanachart@imcinstitute.com

The asset value is now changed



thanachart@imcinstitute.com

Developing a new business network

Thanachart Numnonda, thanachart@imcinstitute.c

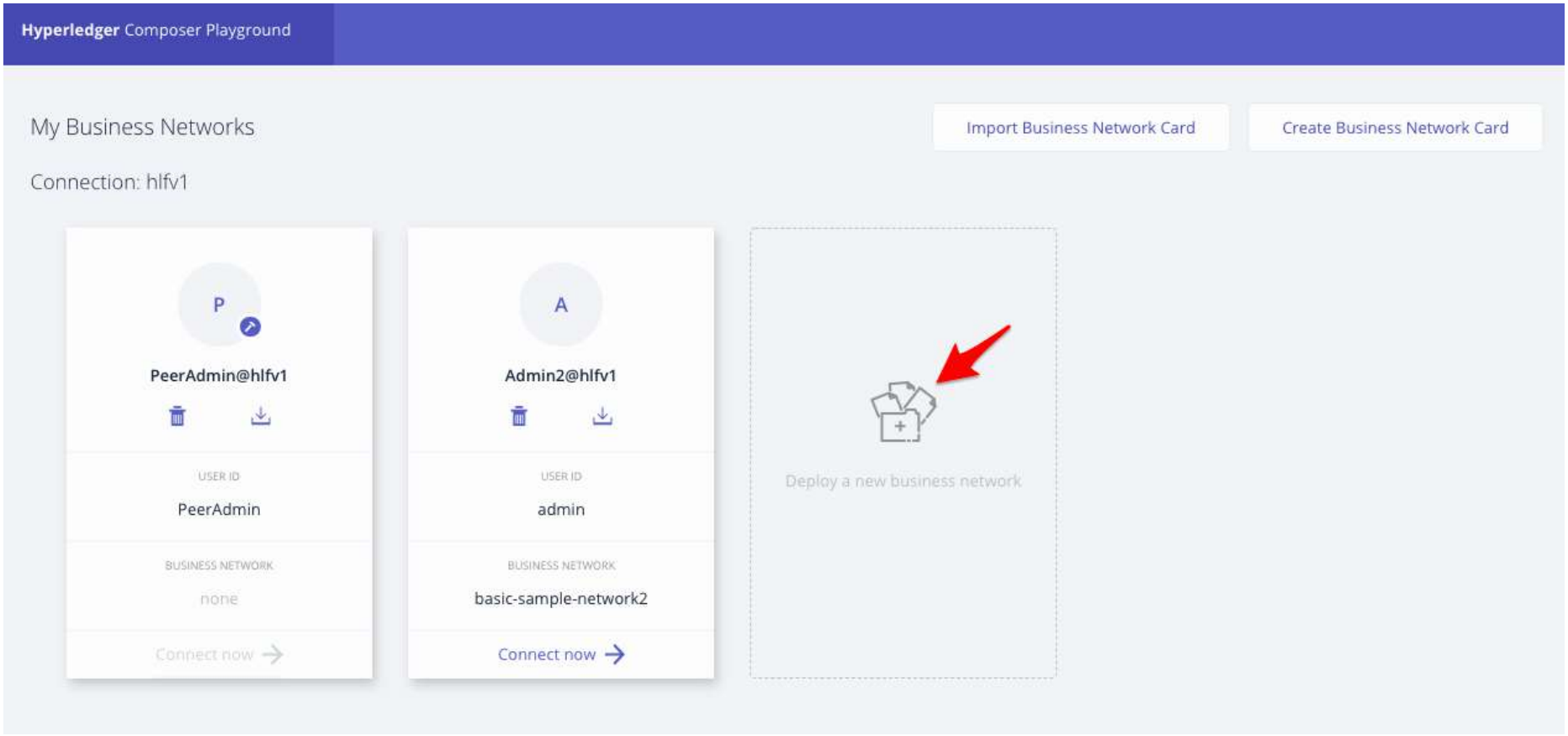
my-bank-network

Definition

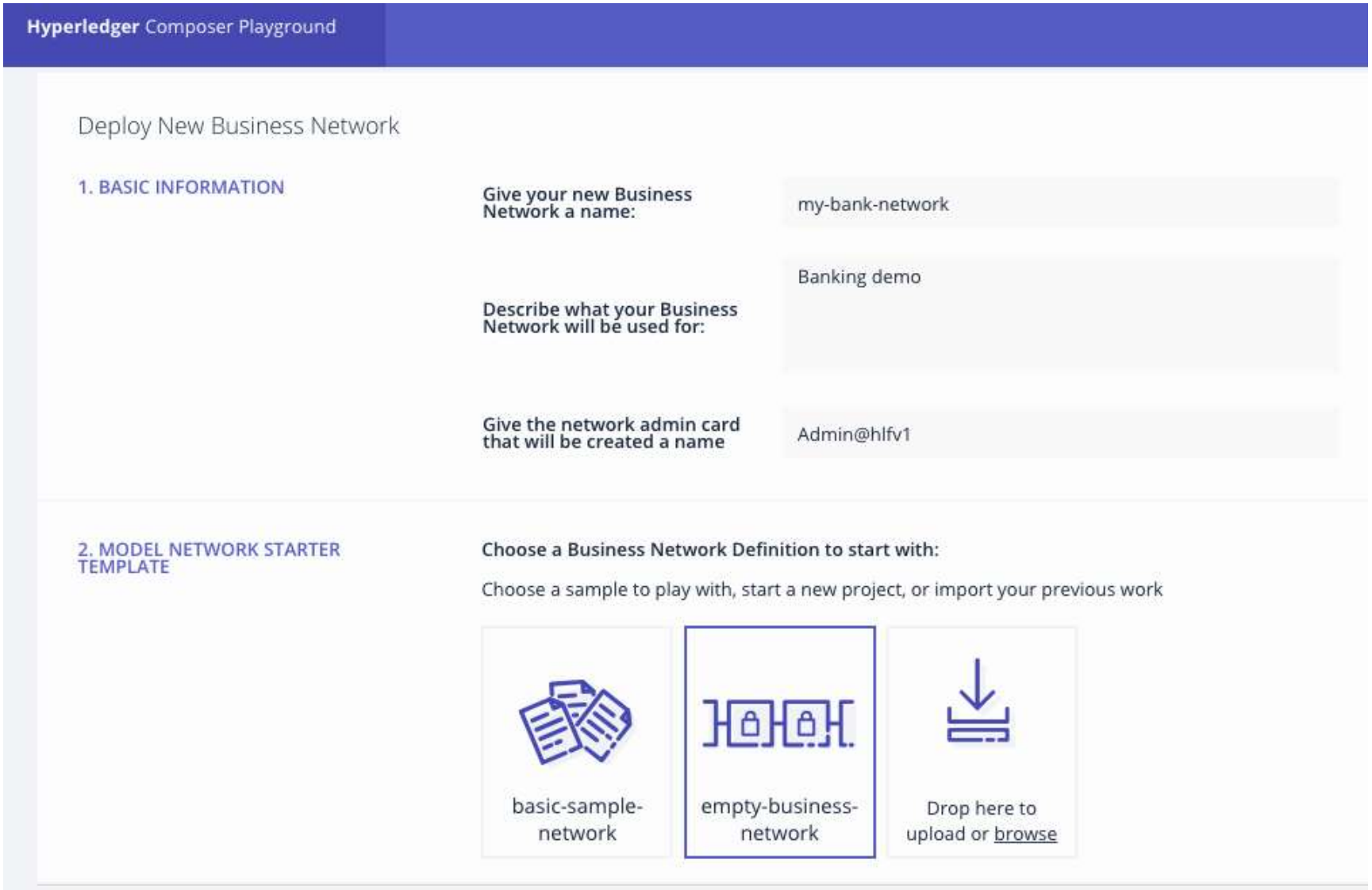
- Asset
  - Account
- Participant
  - Customer
- Transaction
  - AccountTransfer

thanachart@imcinstitute.com

Click Deploy a new business network

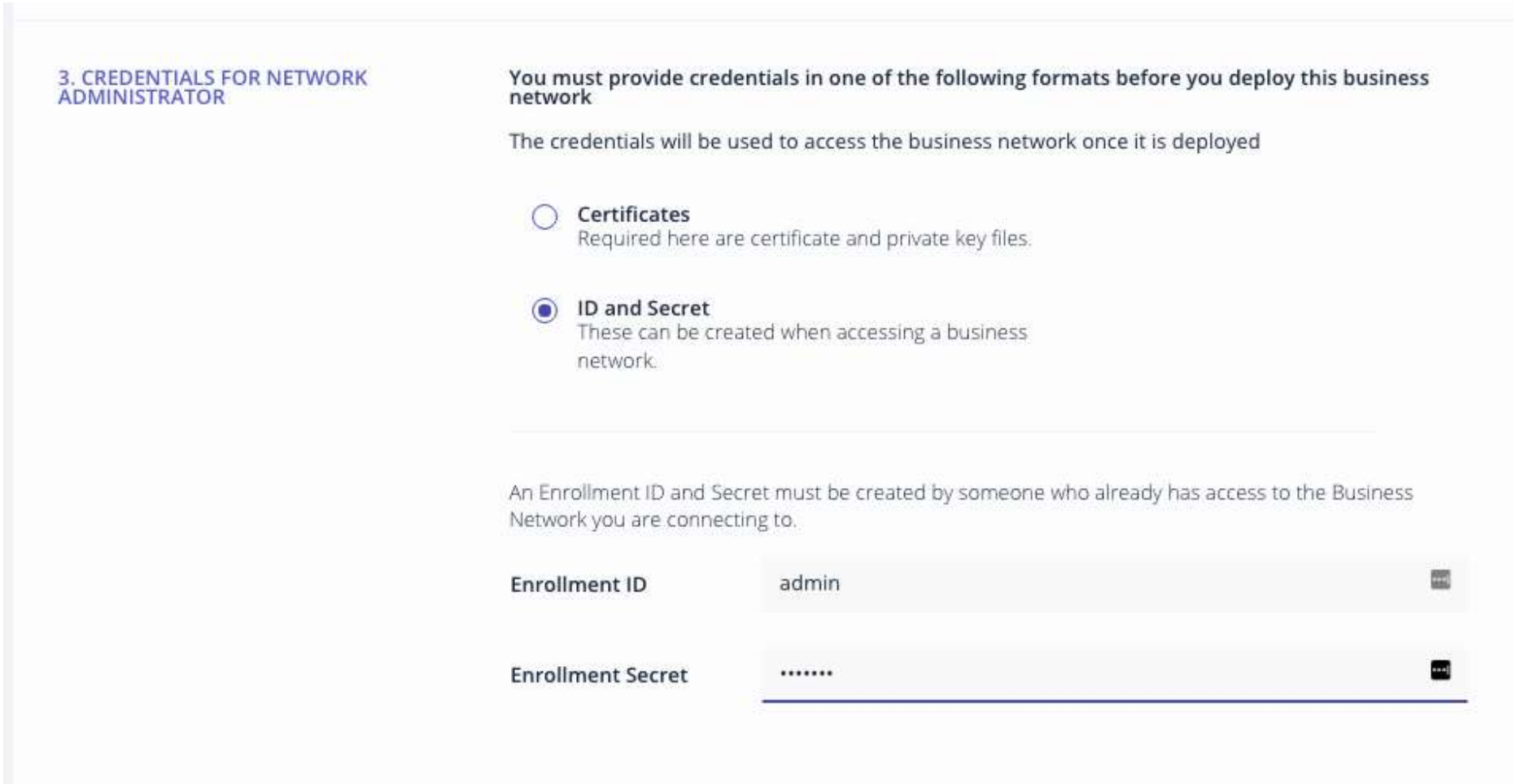


Provide basic information and select empty-business-network (provide the network admin as Admin@hlfv1)

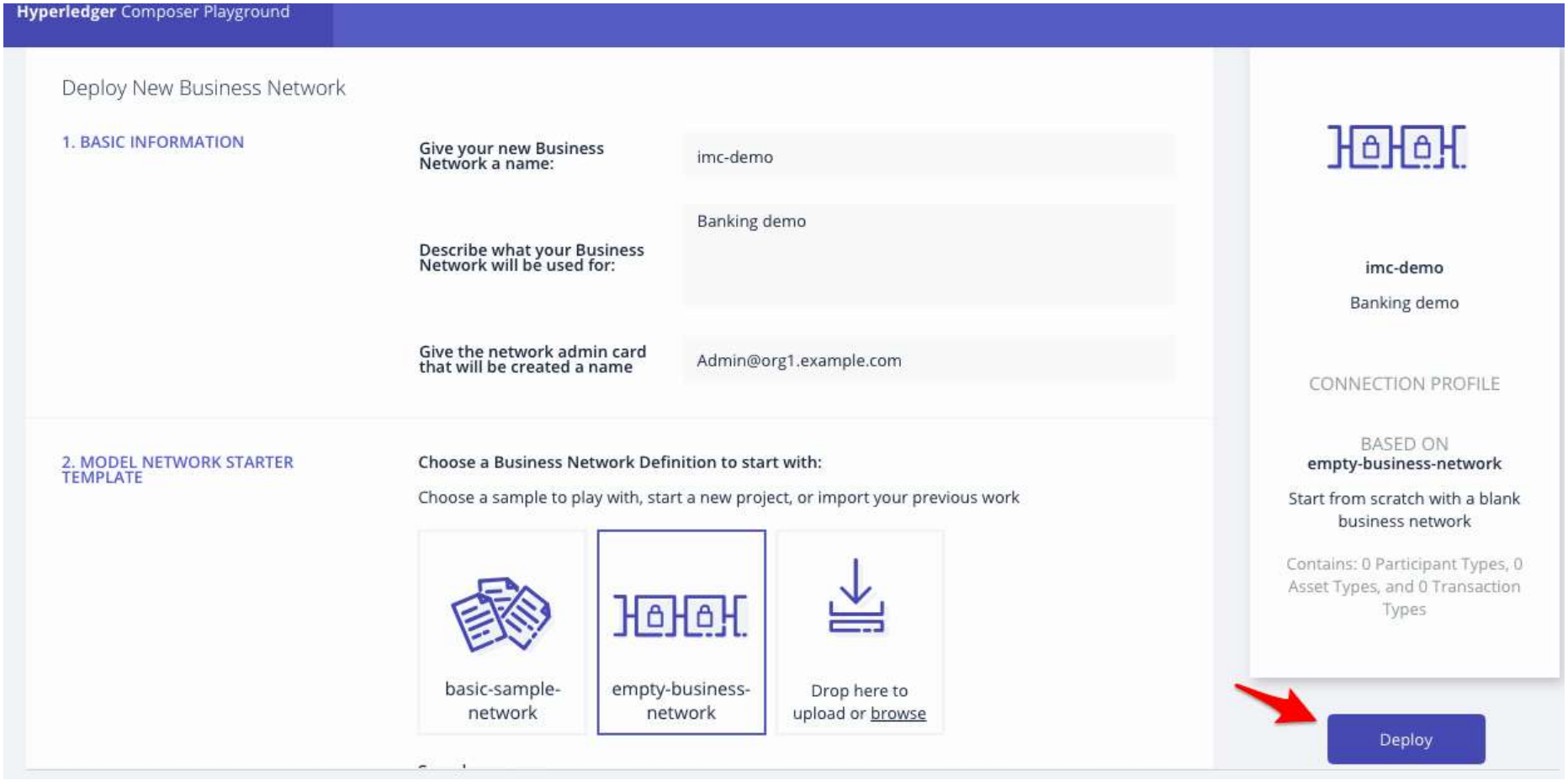


Provide credentials information

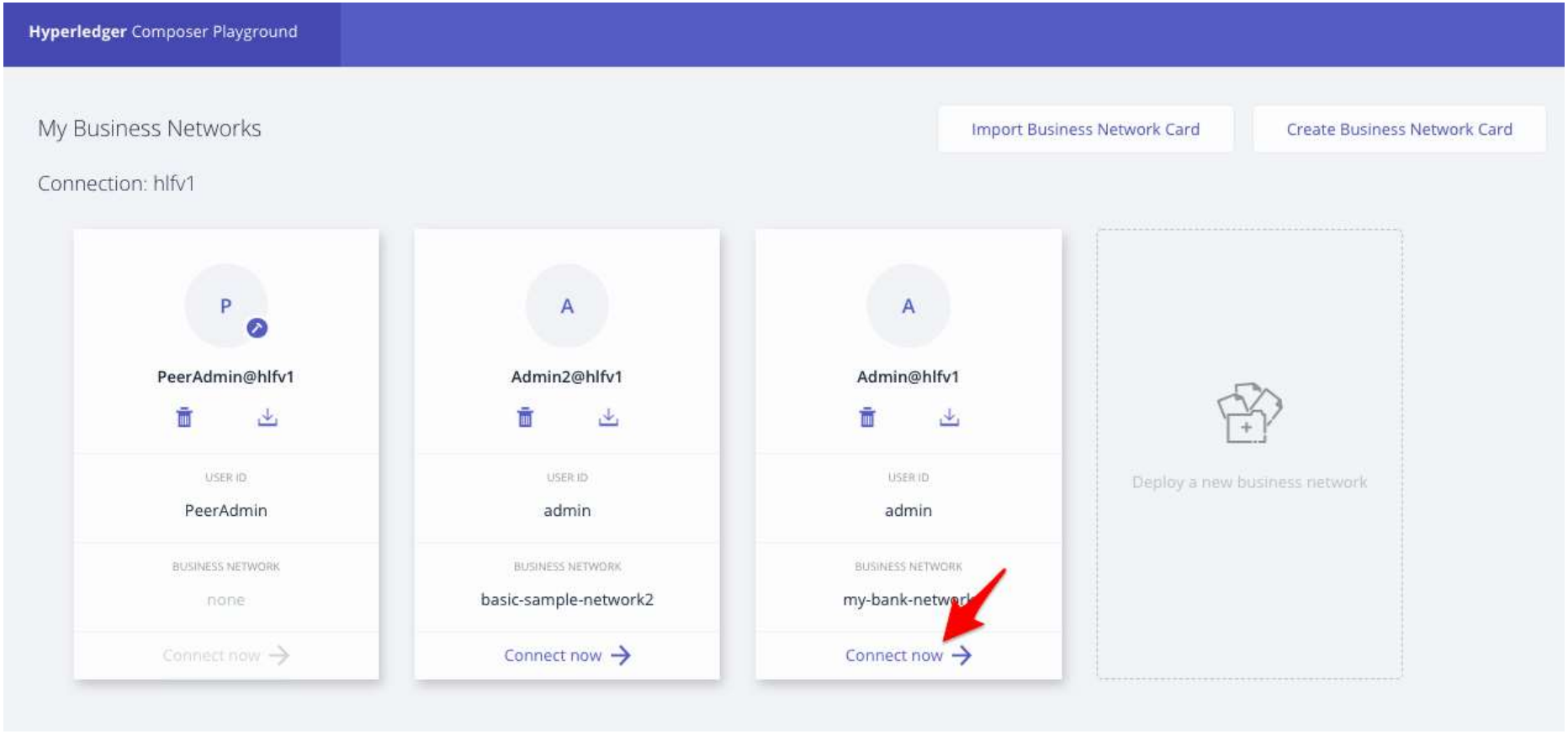
(Assign the Enrolment id: admin Enrolment secret : adminpw)

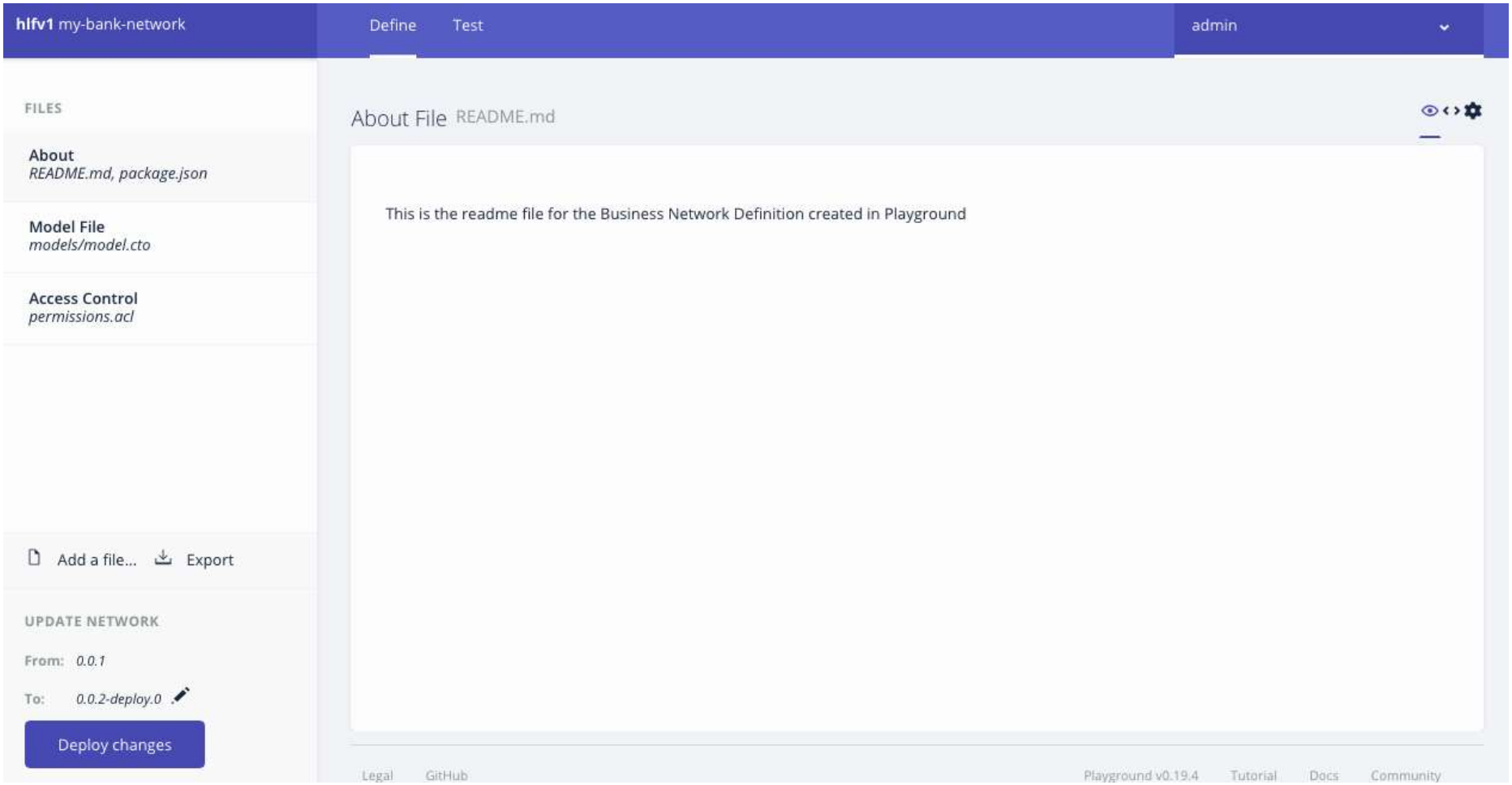


Now the network is defined, click Deploy.



The new business network will be shown, click Connect now





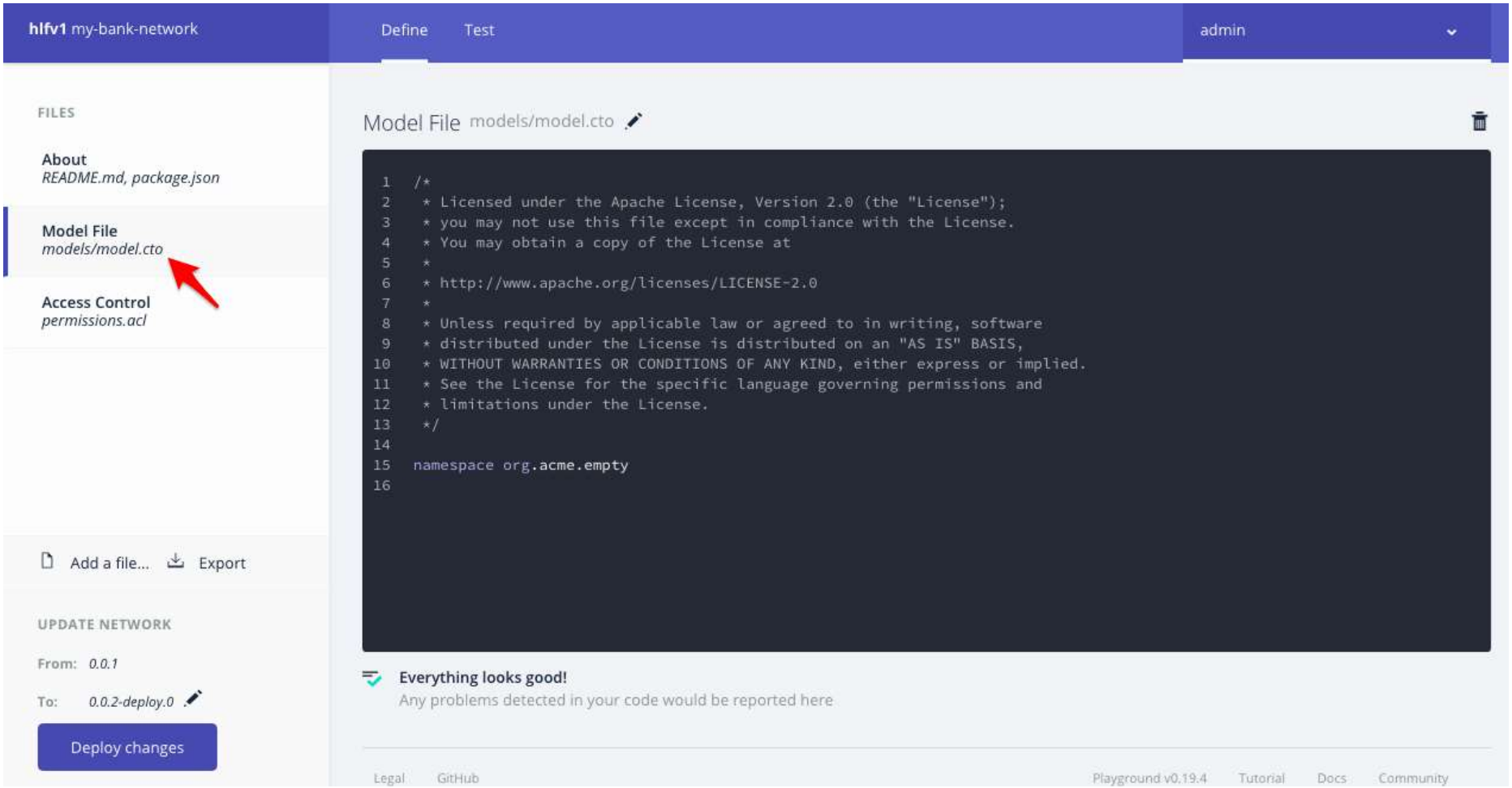
thanachart@imcinstitute.com

Edit a model file

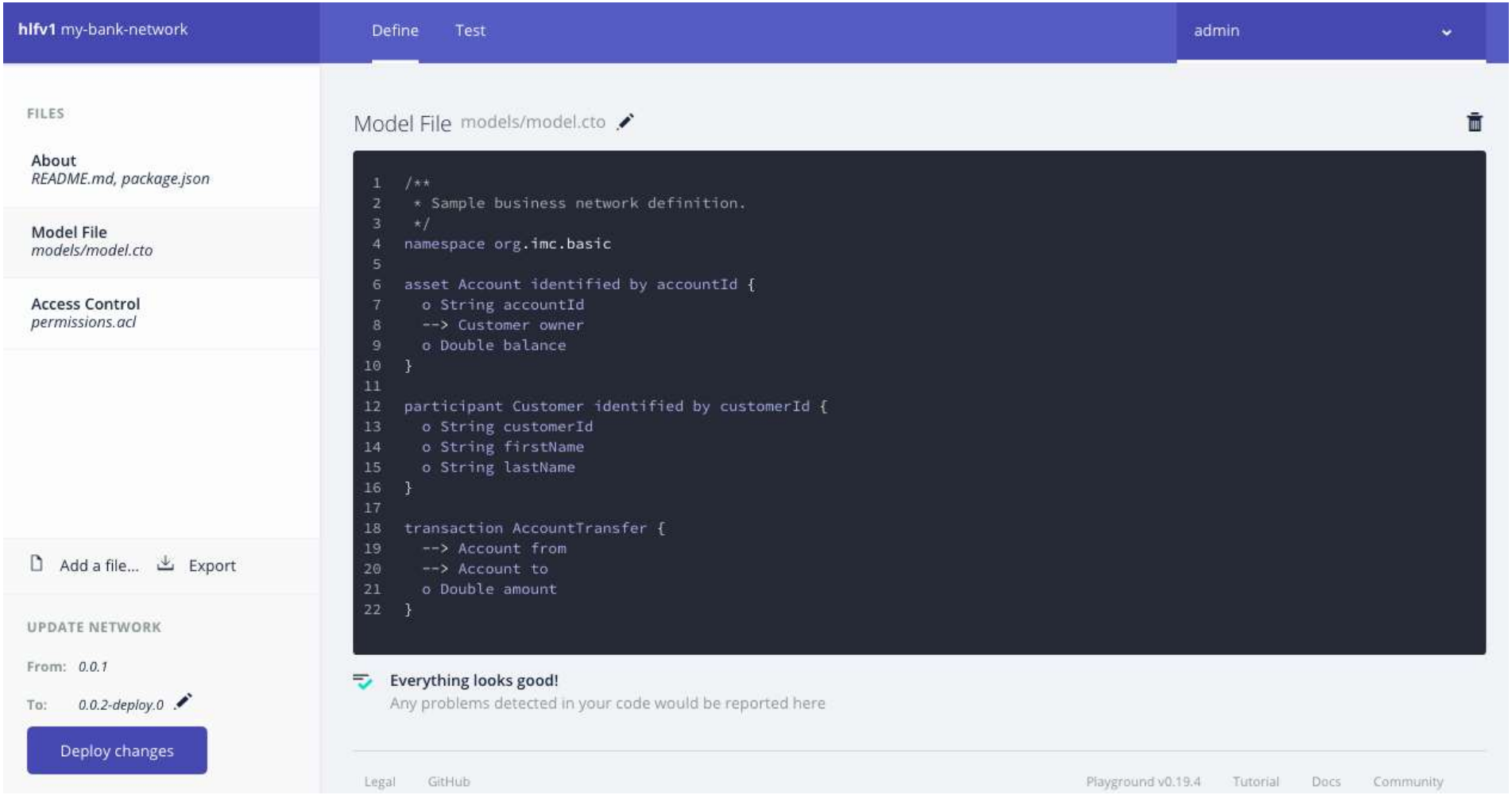
```
/**
 * Sample business network definition.
 */
namespace org.imc.basic

asset Account identified by accountId {
  o String accountId
  --> Customer owner
  o Double balance
}

participant Customer identified by customerId {
  o String customerId
  o String firstName
  o String lastName
}

transaction AccountTransfer {
  --> Account from
  --> Account to
  o Double amount
}
```


thanachart@imcinstitute.com

Select Model file, click edit icon



thanachart@imcinstitute.com



thanachart@imcinstitute.com

Edit a script file

```
/**
* Place a transaction for transfering money
* @param {org.imc.basic.AccountTransfer} AccountTransfer
* @transaction
*/

function accountTransfer(accountTransfer) {

    if (accountTransfer.from.balance < accountTransfer.amount) {
      throw new Error('Insufficient fund');
    }

    accountTransfer.from.balance -= accountTransfer.amount;
    accountTransfer.to.balance += accountTransfer.amount;

    return getAssetRegistry('org.imc.basic.Account')
          .then(function(assetRegistry) {
              return assetRegistry.update(accountTransfer.from);
        })

```

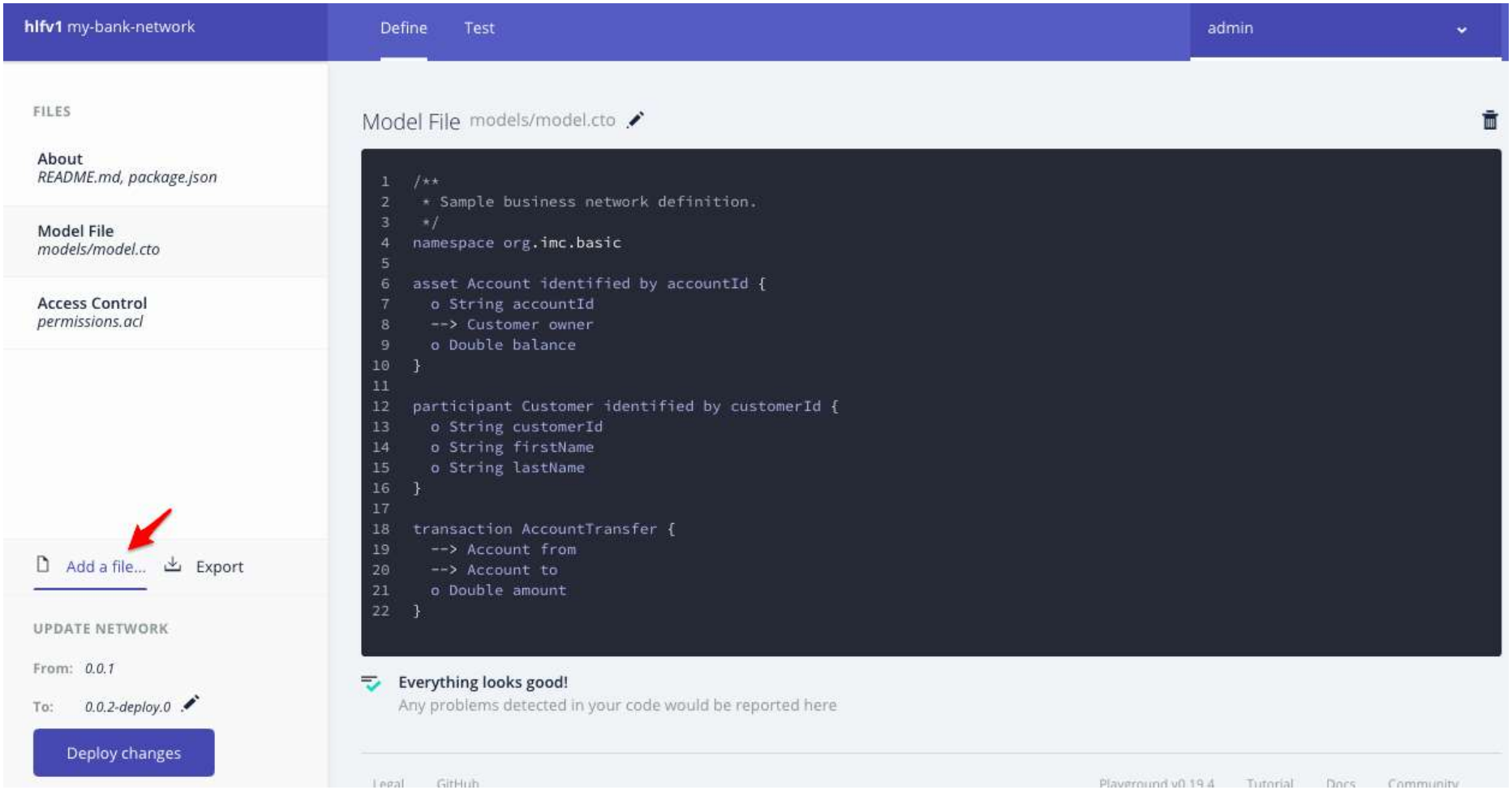

thanachart@imcinstitute.com

Edit a script file (cont.)

```
        .then(function() {
            return getAssetRegistry('org.imc.basic.Account');
        })
        .then(function(assetRegistry) {
            return assetRegistry.update(accountTransfer.to);
        });
}
```

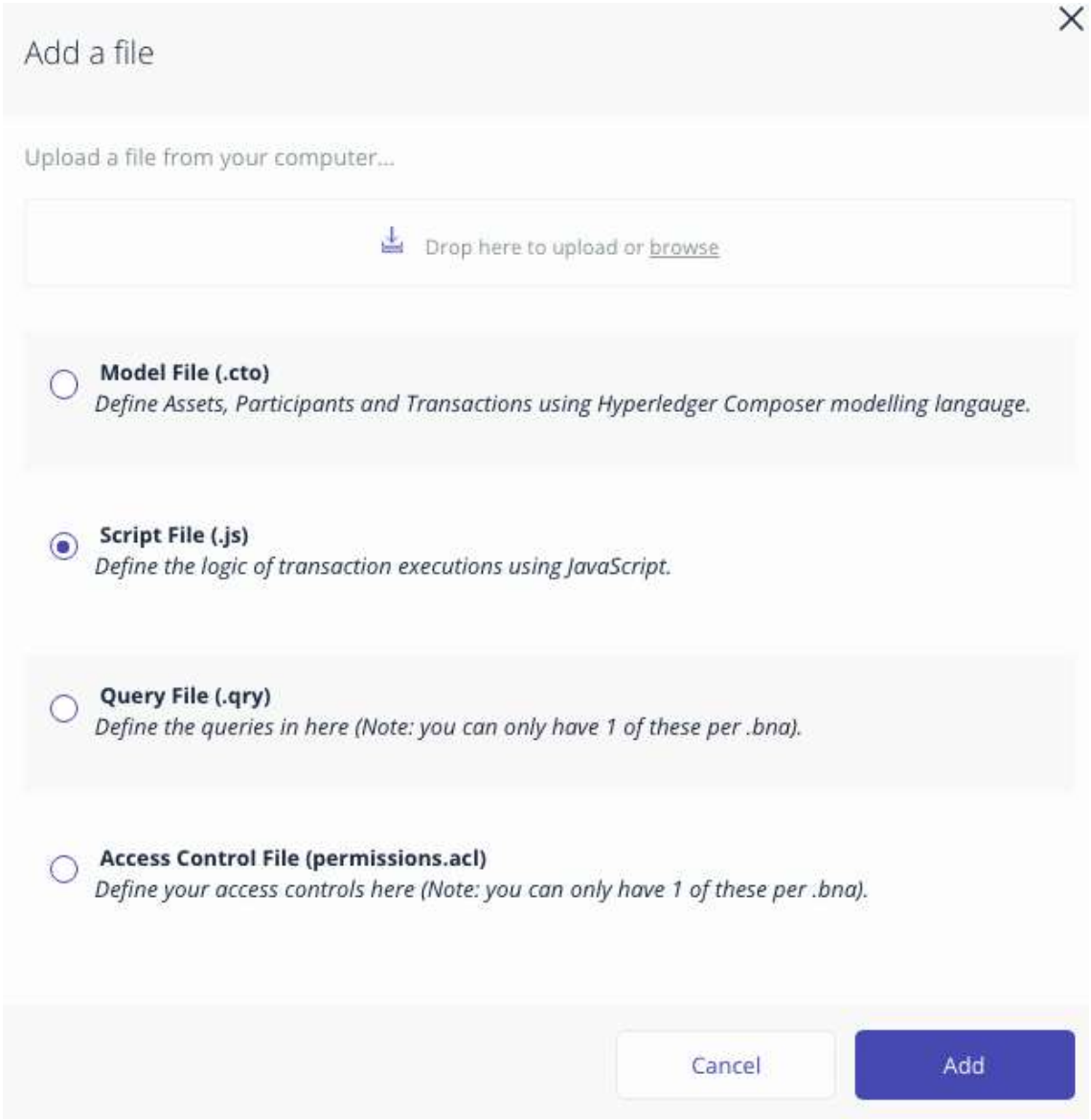

thanachart@imcinstitute.com

Select Add a file



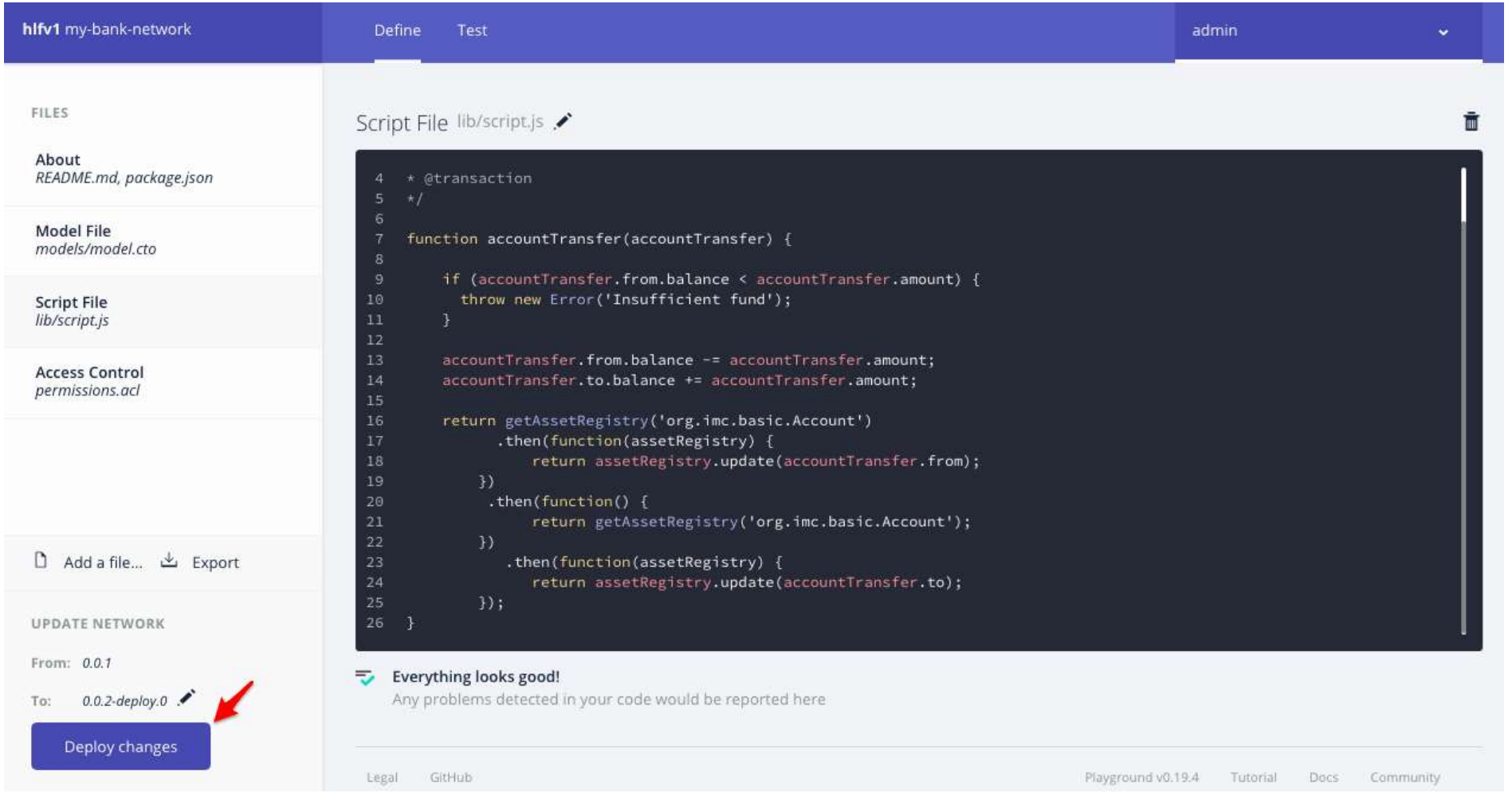
thanachart@imcinstitute.com

Choose Script File(.js), the click Add



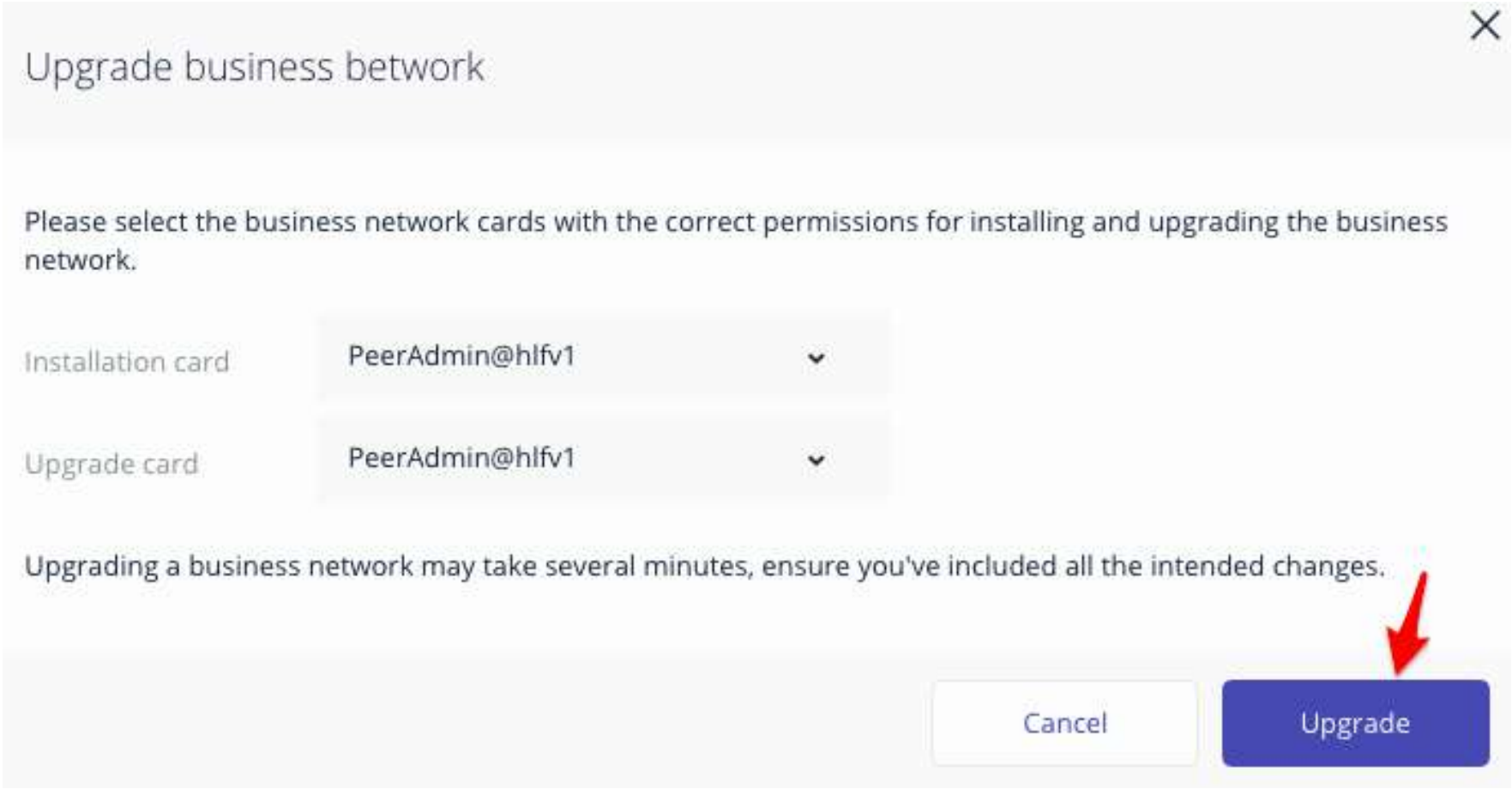
thanachart@imcinstitute.com

Deploying the updated business network, Click Deploy change



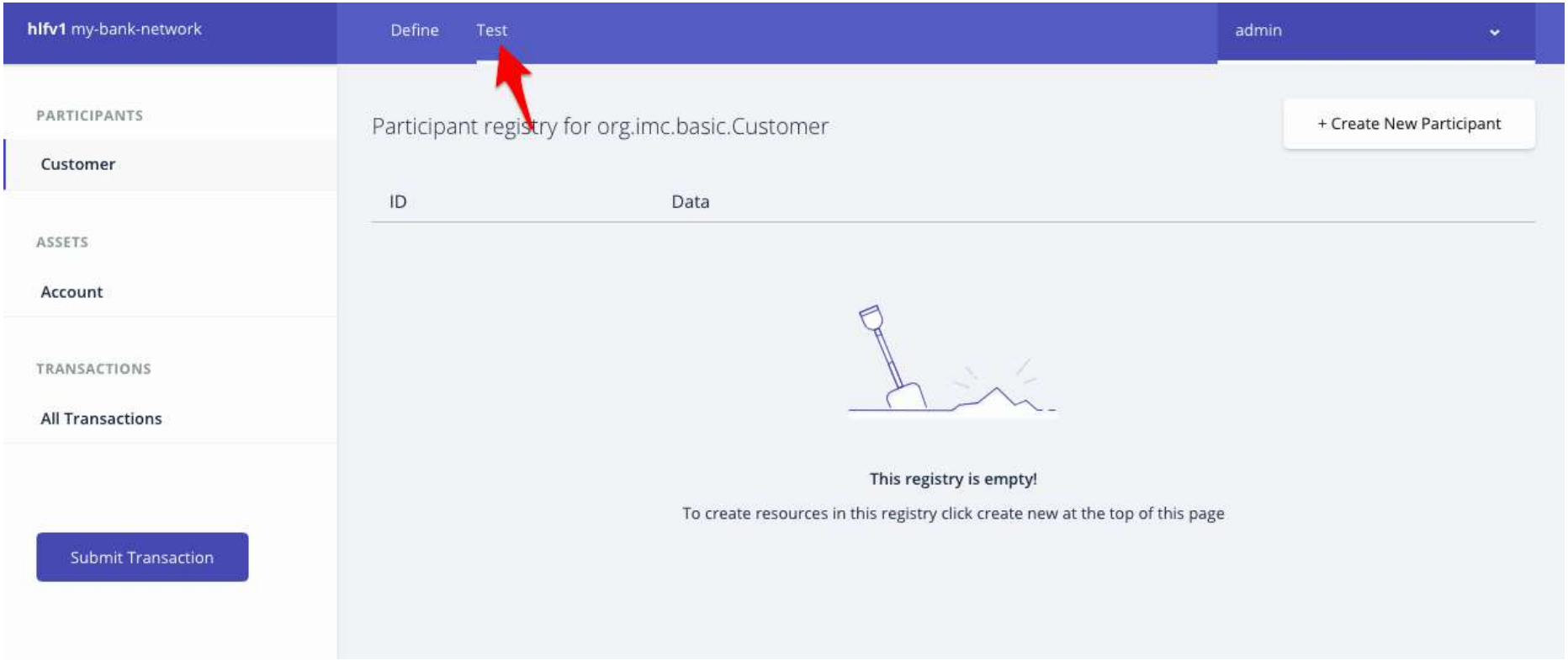
thanachart@imcinstitute.com

Click Upgrade



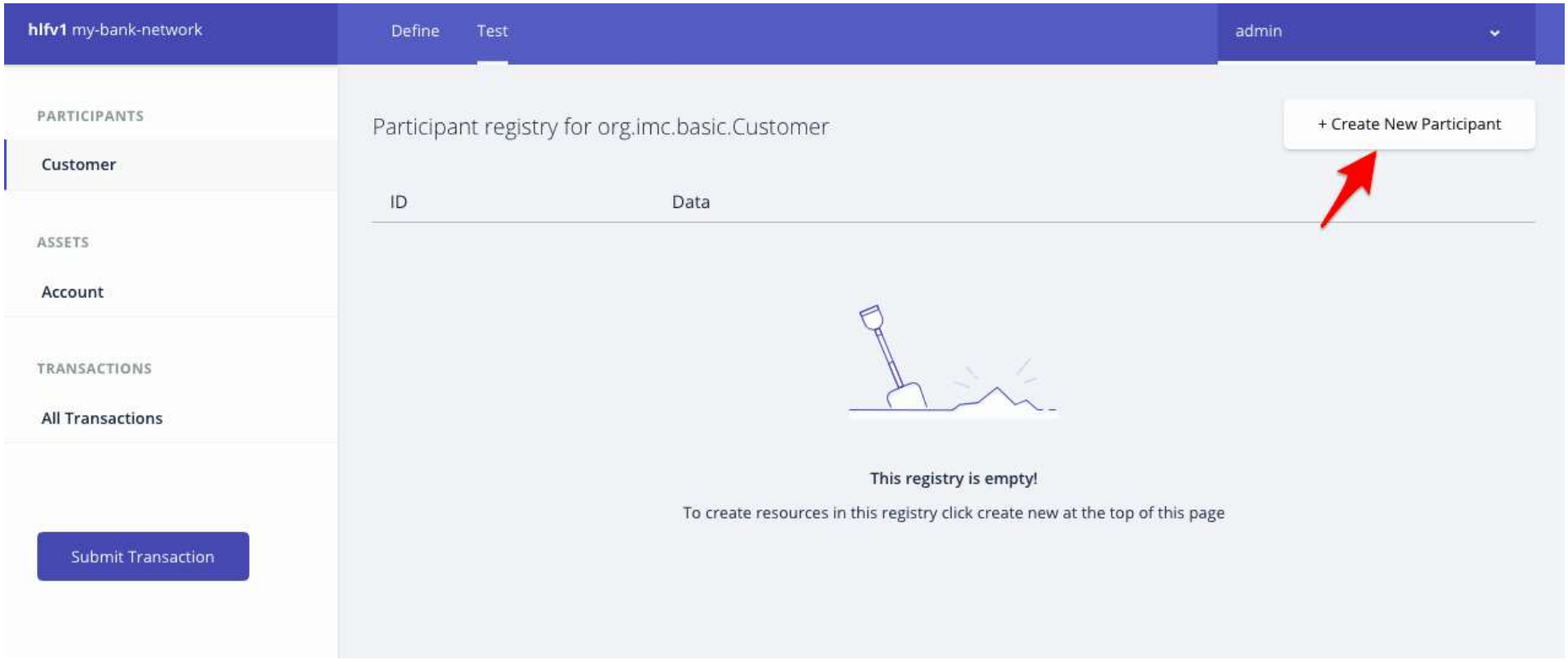
thanachart@imcinstitute.com

Testing the business network definition, Click Test tab



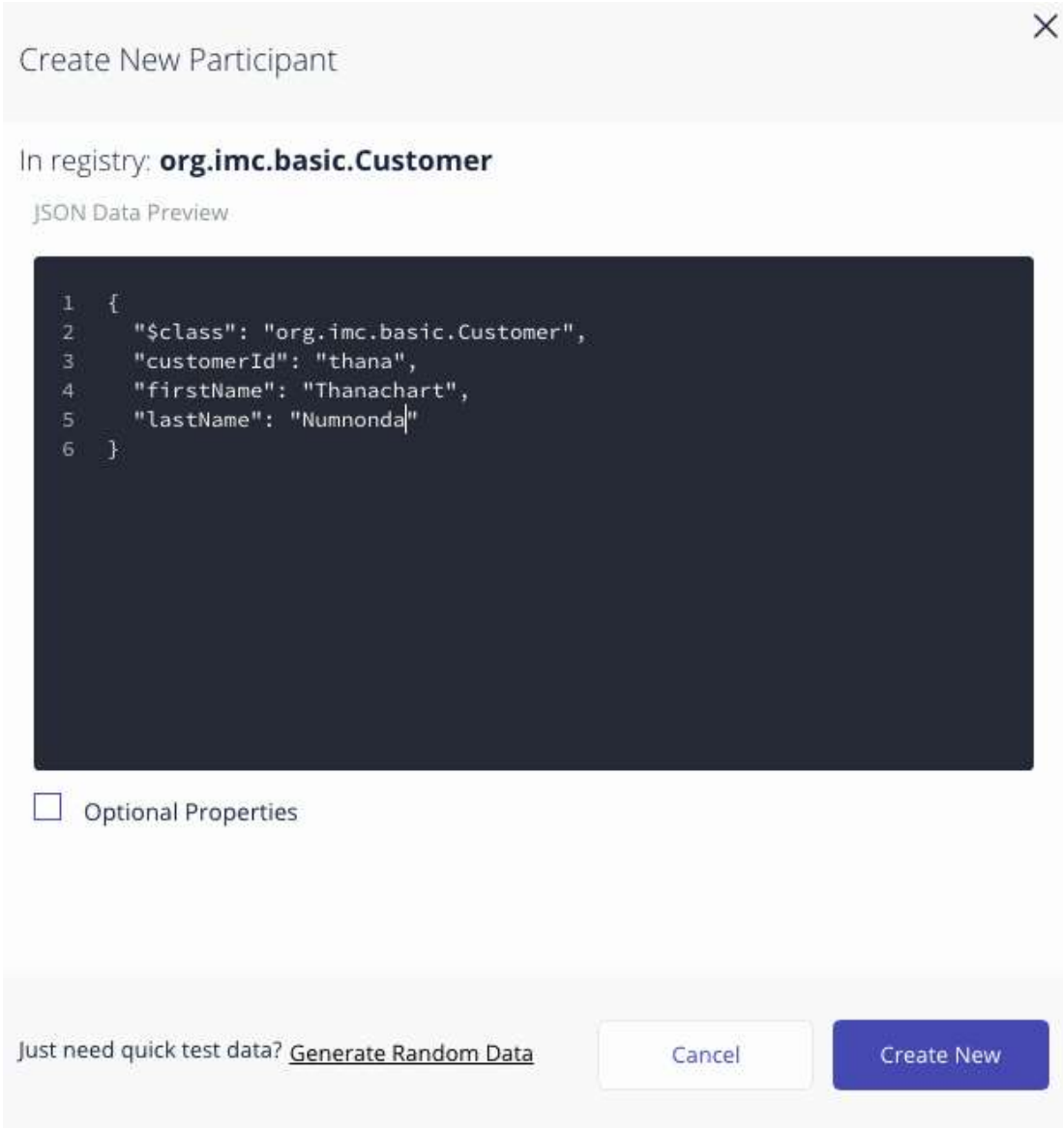
thanachart@imcinstitute.com

Click Create New Participant



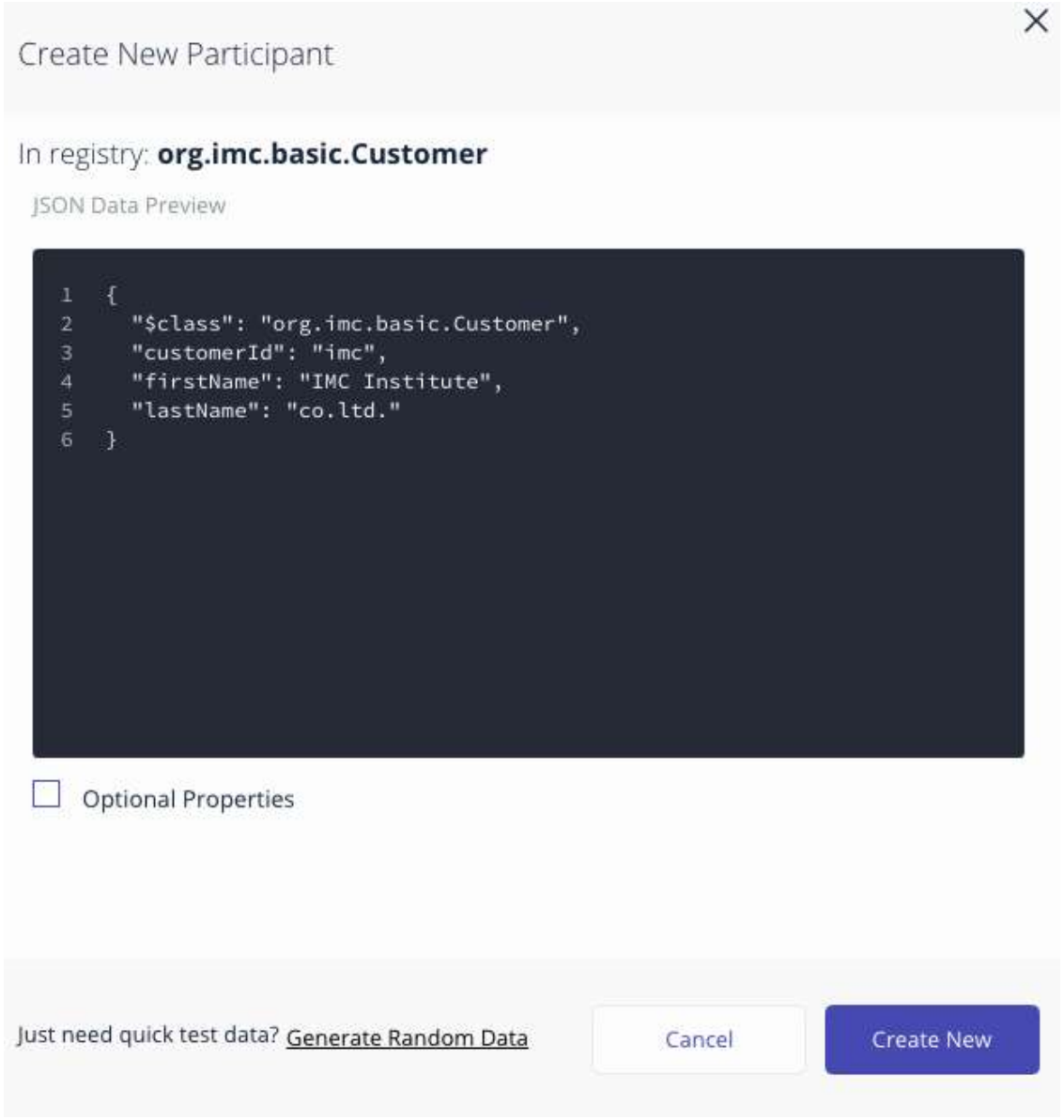
thanachart@imcinstitute.com

Enter the first participant, the click Create New



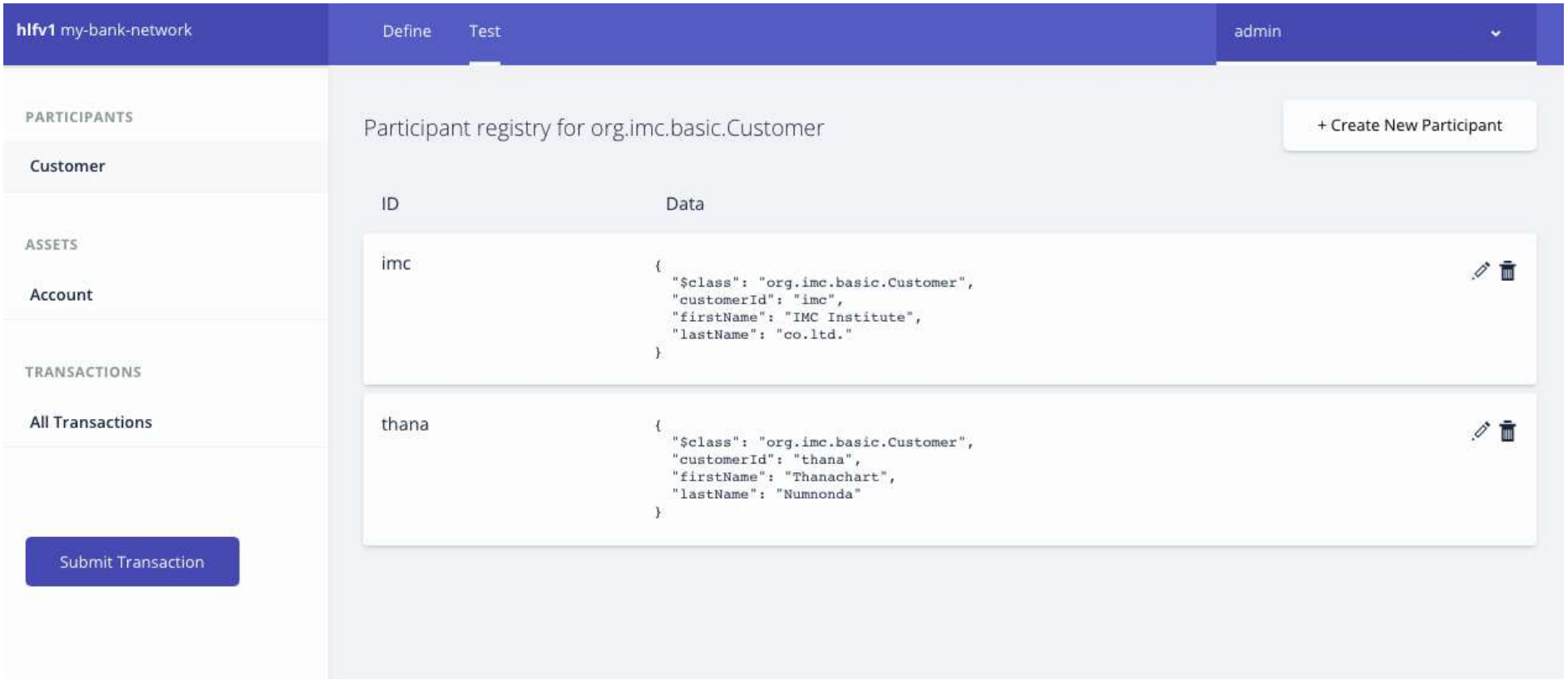
thanachart@imcinstitute.com

Enter the second participant, the click Create New



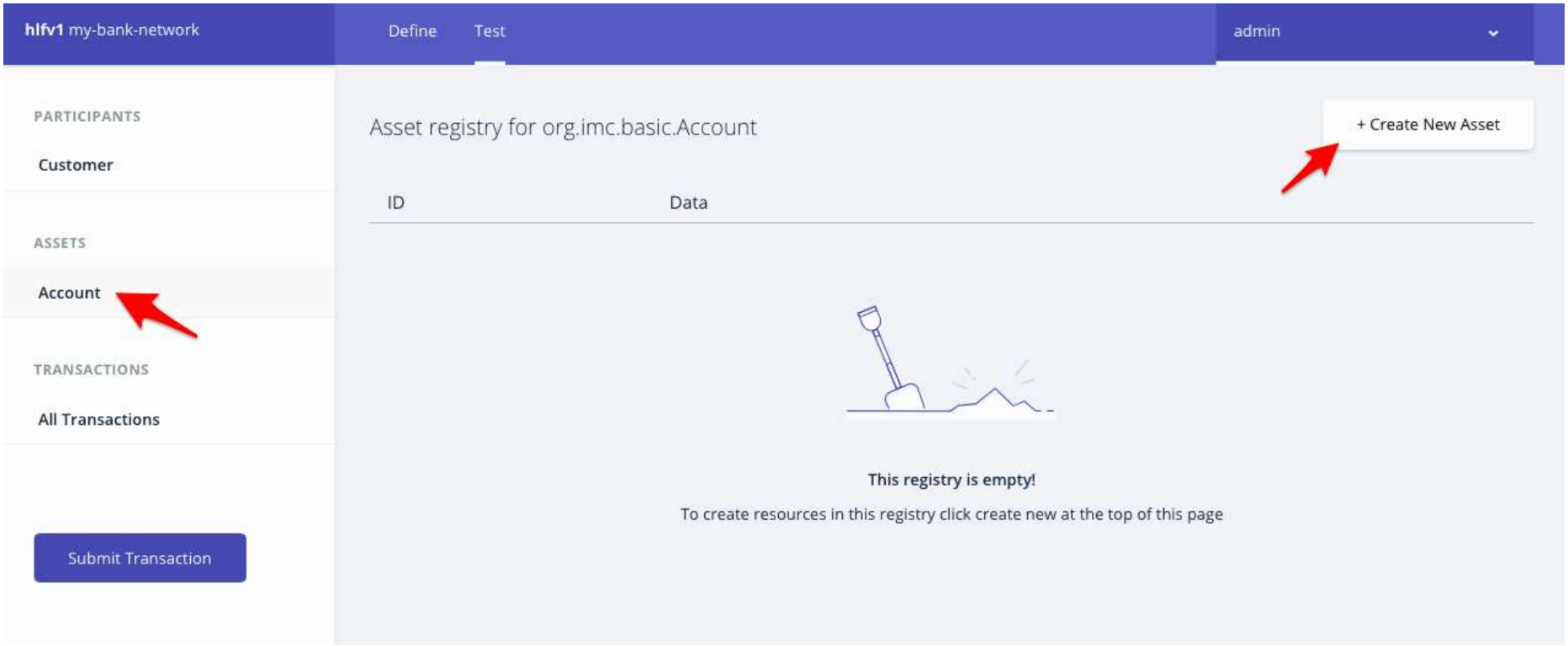
thanachart@imcinstitute.com

A list of participants will be shown



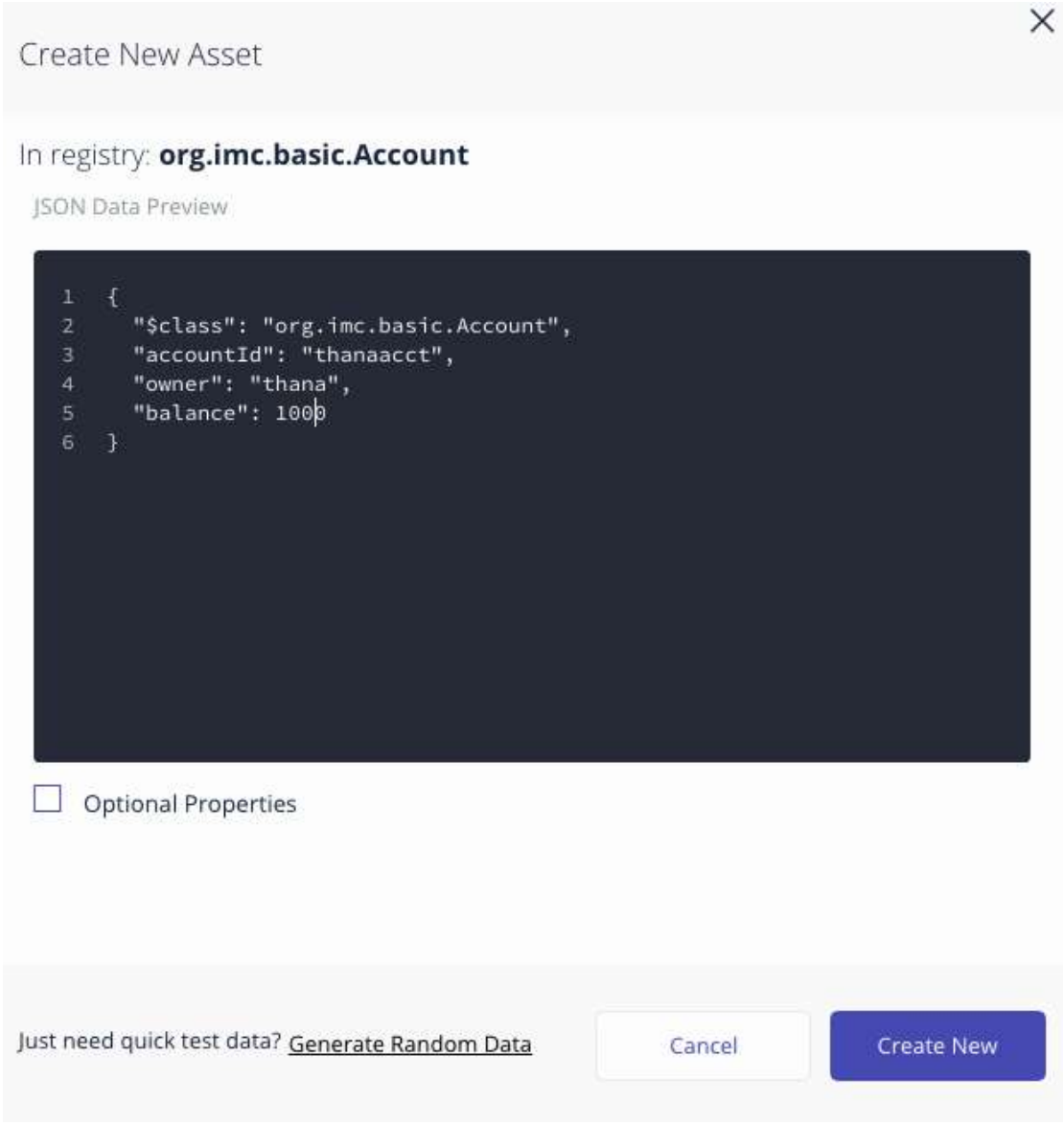
thanachart@imcinstitute.com

Select Account, the click Create New Asset



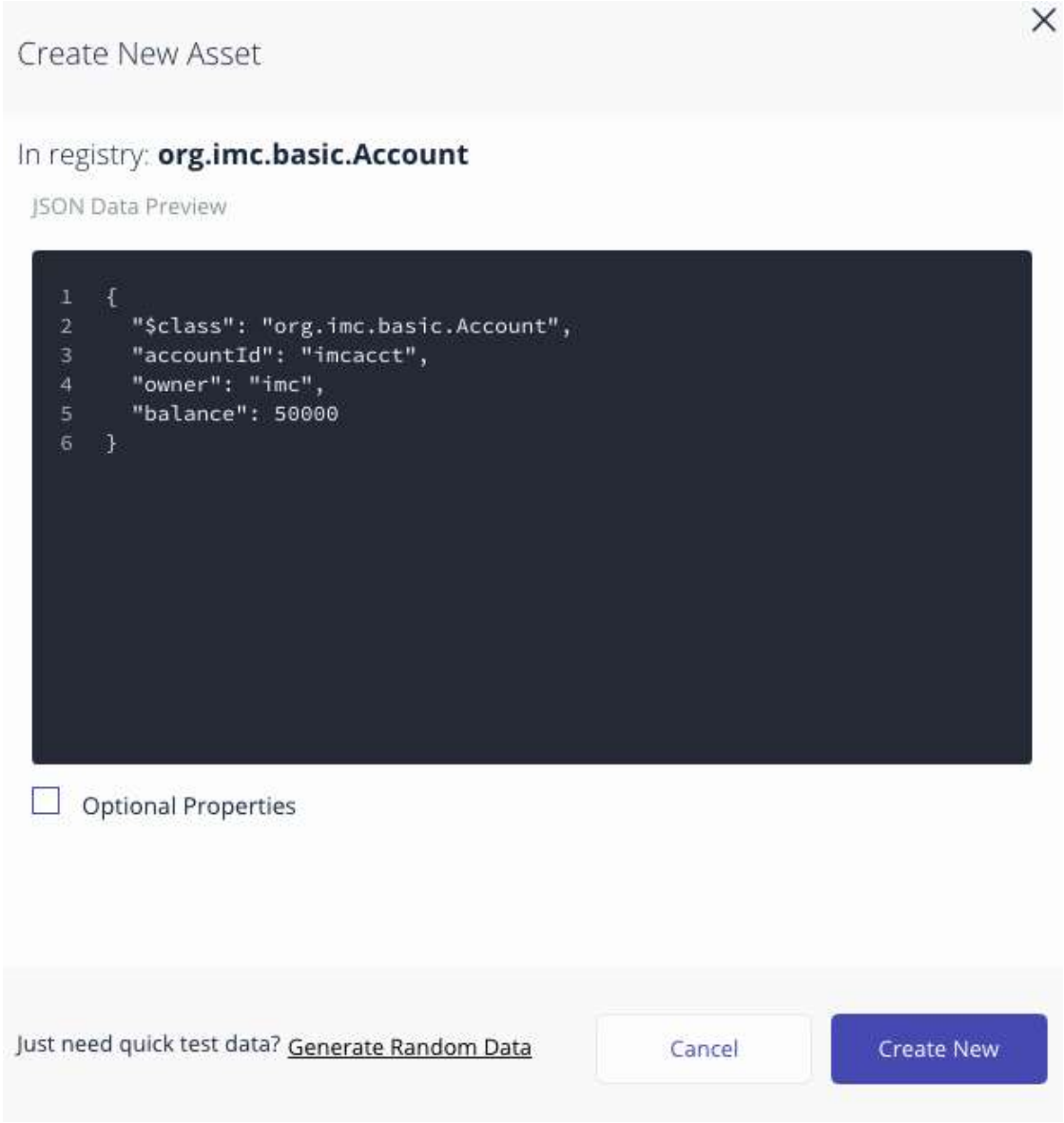
thanachart@imcinstitute.com

Enter the first account, the click Create New



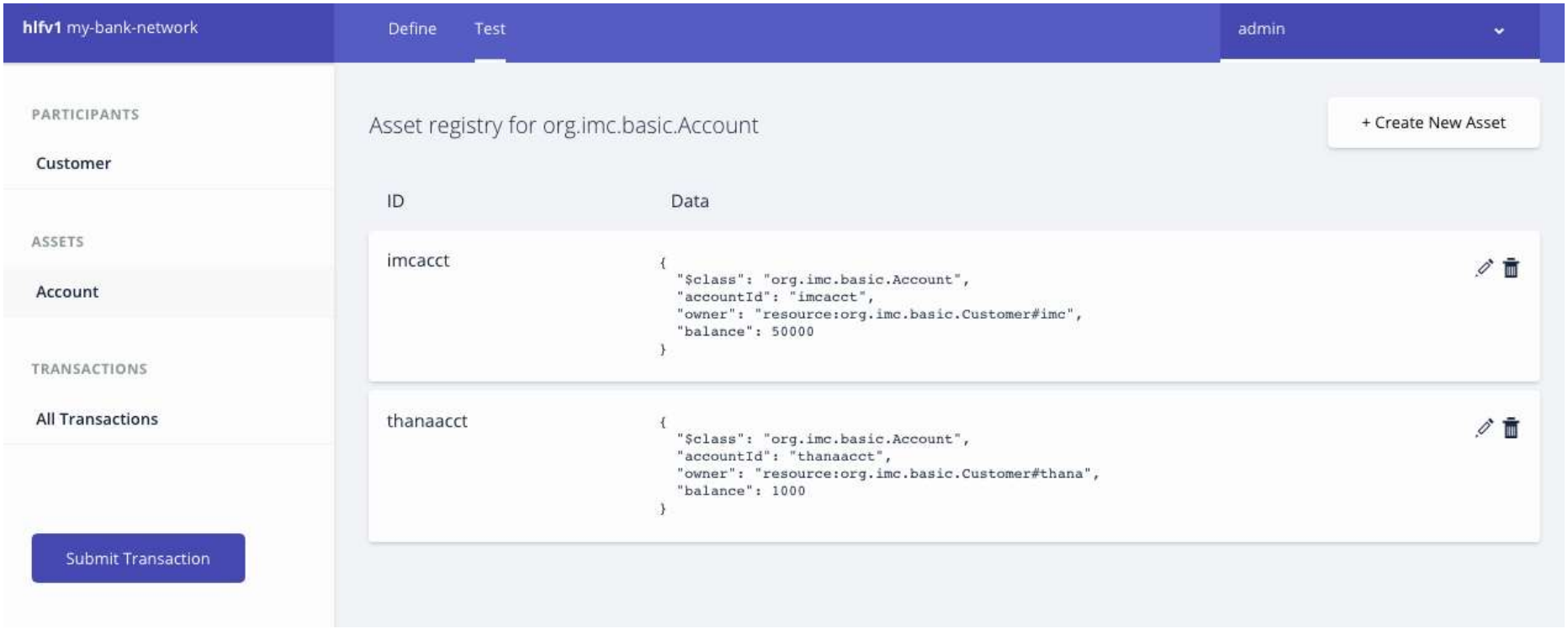
thanachart@imcinstitute.com

Enter the second account, the click Create New



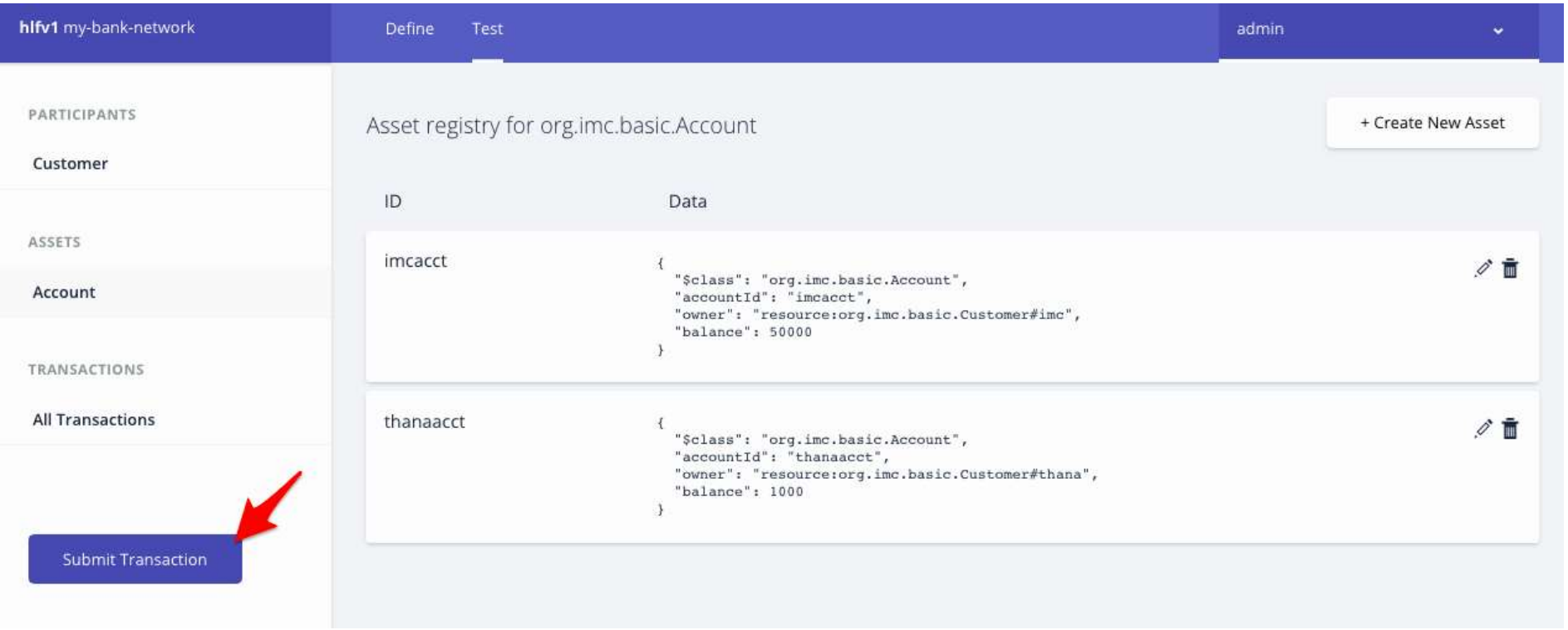
thanachart@imcinstitute.com

A list of accounts will be shown



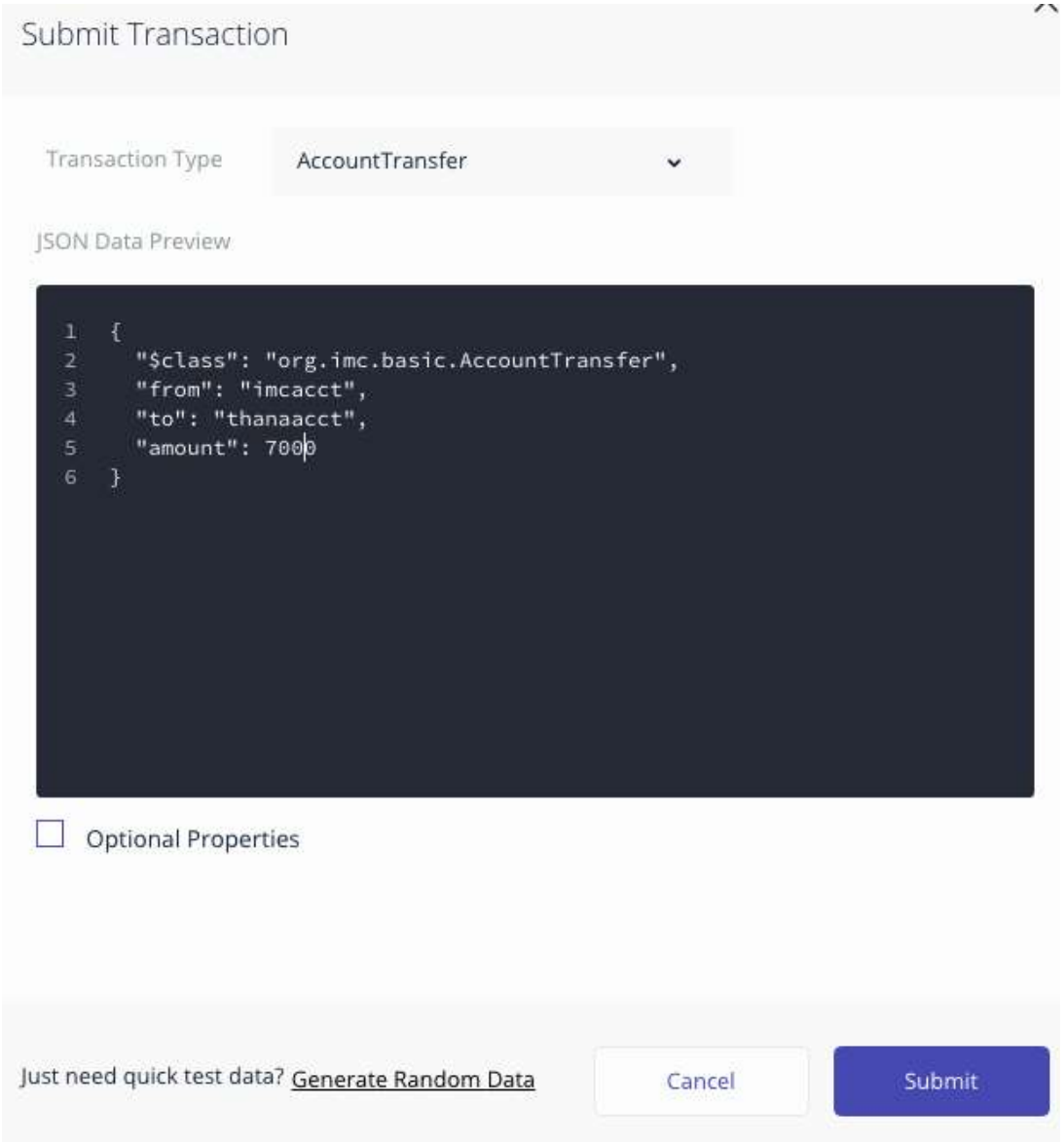
thanachart@imcinstitute.com

Click Submit Transaction



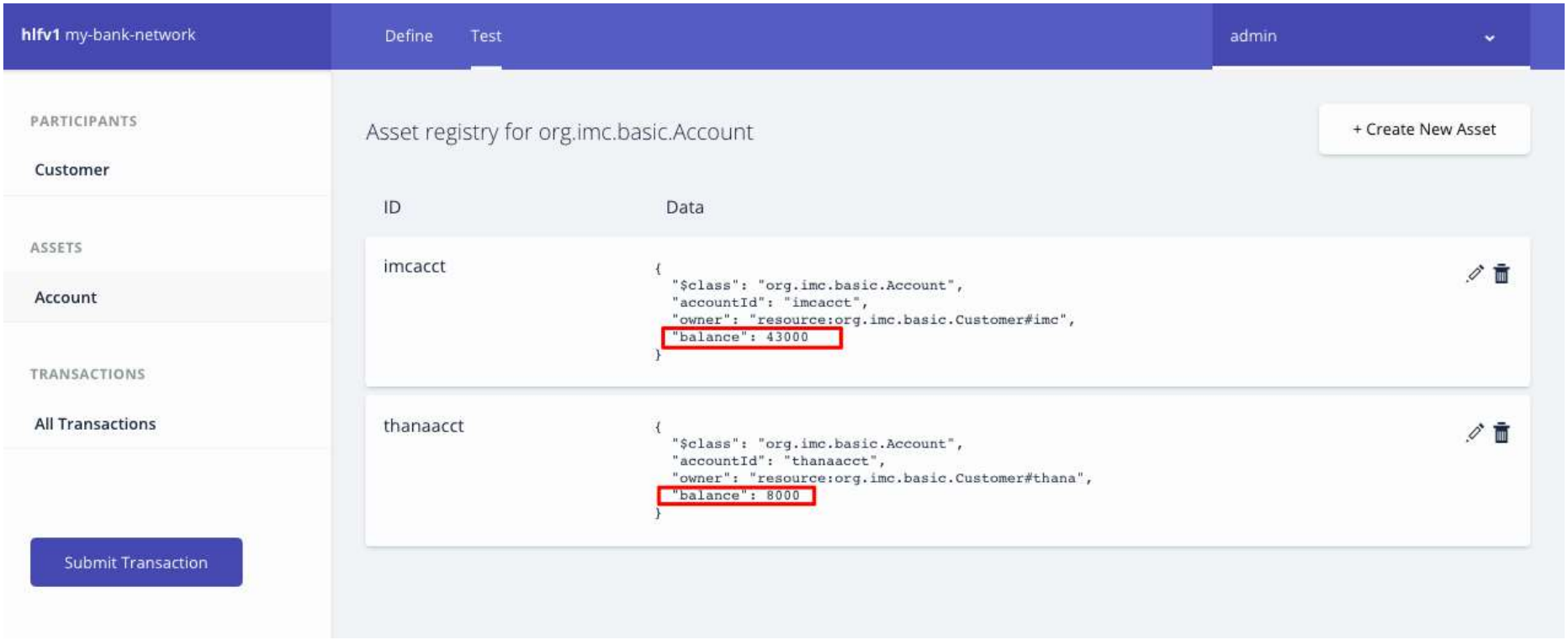
thanachart@imcinstitute.com

Enter the transaction information, the click Submit



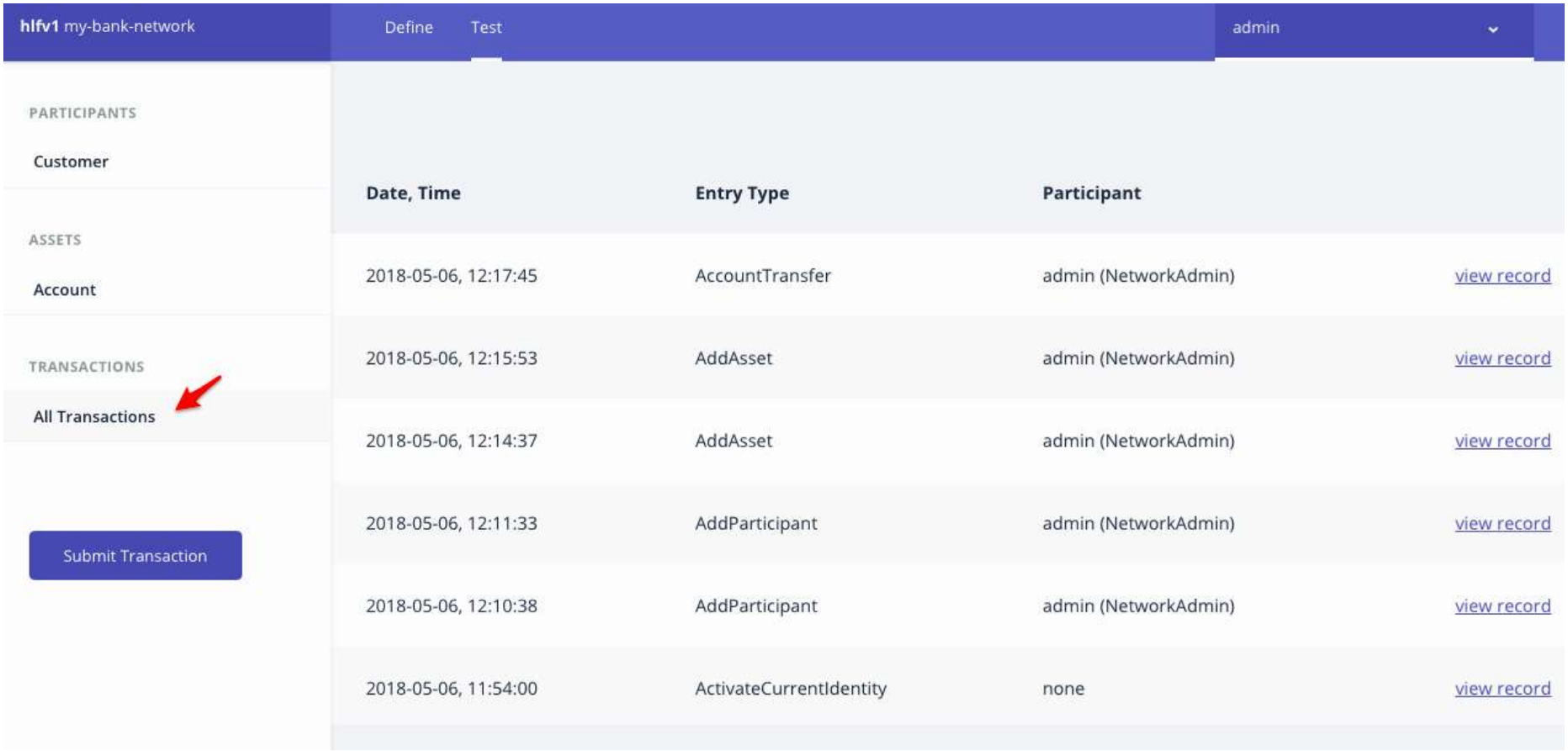
thanachart@imcinstitute.com

The account balances are now changed



thanachart@imcinstitute.com

All transactions show events



thanachart@imcinstitute.com

Generate REST APIs

Thanachart Numnonda, thanachart@imcinstitute.c

List all of the business cards

$ composer card list

```
thanachart@imchyperledger:~$ composer card list
The following Business Network Cards are available:

Connection Profile: hlfv1
```

Card Name	UserId	Business Network
Admin@hlfv1	admin	my-bank-network
Admin2@hlfv1	admin	basic-sample-network2
PeerAdmin@hlfv1	PeerAdmin	

```
Issue composer card list --card <Card Name> to get details a specific card

Command succeeded
```


thanachart@imcinstitute.com

Start the REST server & generate the API

$ composer-rest-server

Enter the name of the business network card to use: Admin@hlfv1
Specify if you want namespaces in the generated REST API: never use namespaces
Specify if you want to use an API key to secure the REST API: No
Specify if you want to enable authentication for the REST API using Passport: No
Specify if you want to enable event publication over WebSockets: Yes
Specify if you want to enable TLS security for the REST API: No

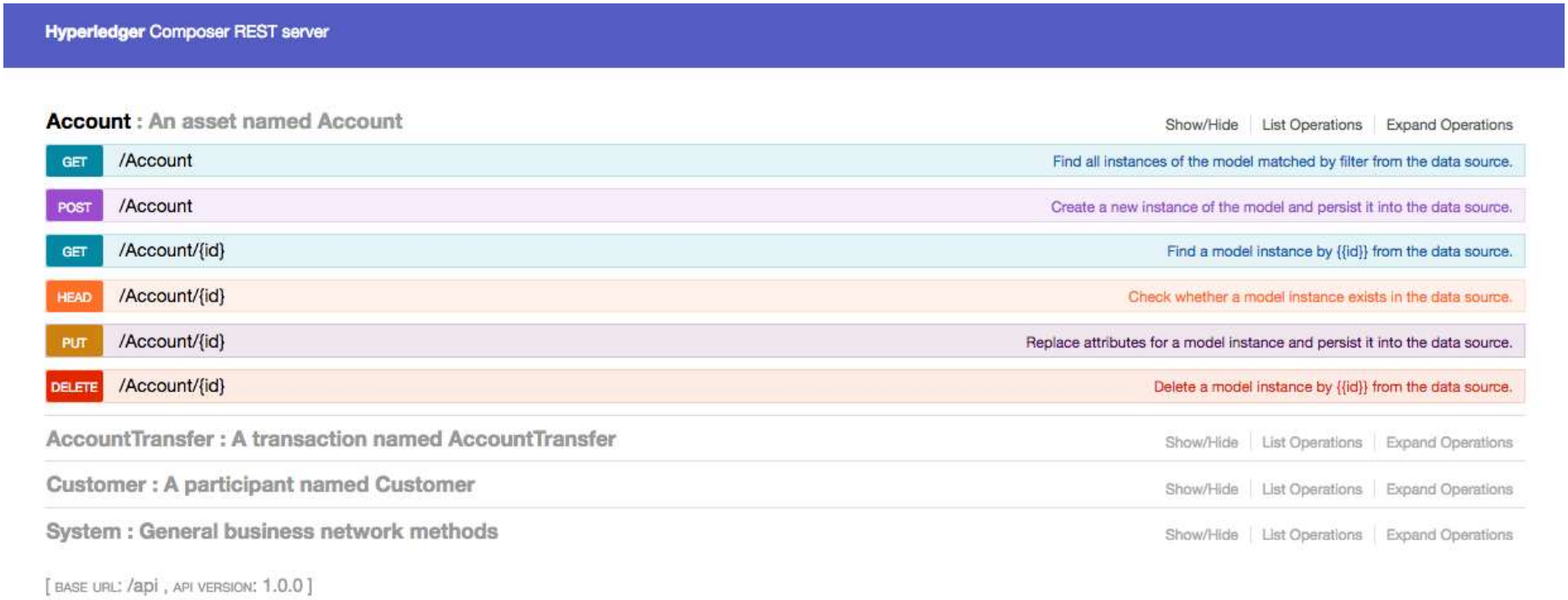
thanachart@imcinstitute.com

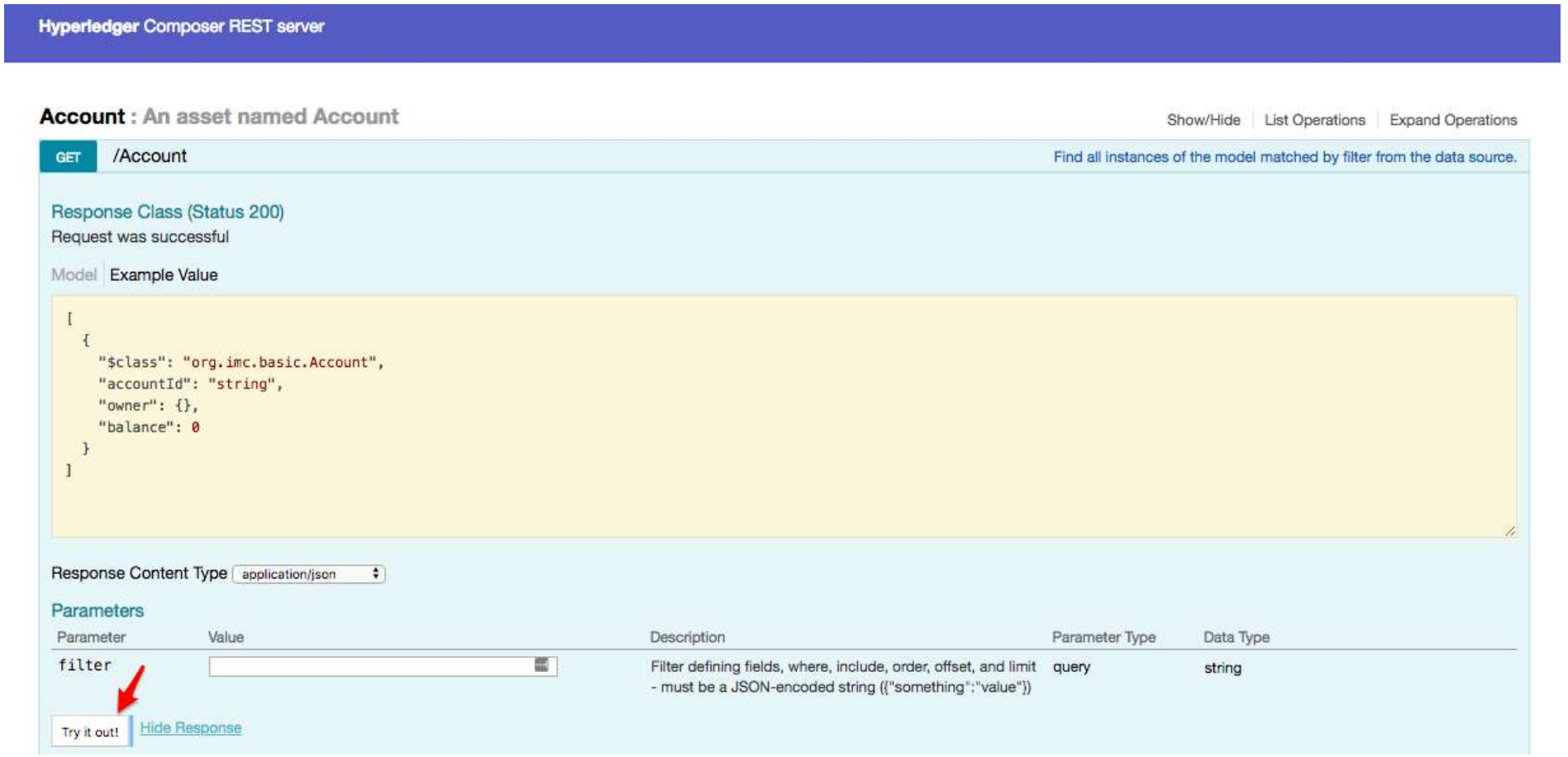
```
thanachart@imchyperledger:~$ composer-rest-server
? Enter the name of the business network card to use: Admin@hlfv1
? Specify if you want namespaces in the generated REST API: never use namespaces
? Specify if you want to use an API key to secure the REST API: No
? Specify if you want to enable authentication for the REST API using Passport: No
? Specify if you want to enable event publication over WebSockets: Yes
? Specify if you want to enable TLS security for the REST API: No

To restart the REST server using the same options, issue the following command:
   composer-rest-server -c Admin@hlfv1 -n never -w true

Discovering types from business network definition ...
Discovered types from business network definition
Generating schemas for all types in business network definition ...
Generated schemas for all types in business network definition
Adding schemas for all types to Loopback ...
Added schemas for all types to Loopback
Web server listening at: http://localhost:3000
Browse your REST API at http://localhost:3000/explorer
```


thanachart@imcinstitute.com

Browse REST APIs (http://ip-address::3000/explorer)

Test the REST APIs




Hyperledger Composer REST server
Account : An asset named Account
Show/Hide
List Operations
Expand Operations
GET
/Account
Find all instances of the model matched by filter from the data source.
Response Class (Status 200)
Request was successful
Model
Example Value
[
{
"$class": "org.imc.basic.Account",
"accountId": "string",
"owner": {},
"balance": 0
}
]
Response Content Type
application/json
Parameters
Parameter
Value
Description
Parameter Type
Data Type
filter
Filter defining fields, where, include, order, offset, and limit - must be a JSON-encoded string ({"something":"value"})
query
string
Try it out!
Hide Response

Curl

```
curl -X GET --header 'Accept: application/json' 'http://35.226.204.101:3000/api/Account'
```

Request URL

```
http://35.226.204.101:3000/api/Account
```

Response Body


```
[
  {
    "$class": "org.imc.basic.Account",
    "accountId": "imcacct",
    "owner": "resource:org.imc.basic.Customer#imc",
    "balance": 43000
  },
  {
    "$class": "org.imc.basic.Account",
    "accountId": "thanaacct",
    "owner": "resource:org.imc.basic.Customer#thana",
    "balance": 8000
  }
]
```


Response Code

```
200
```

Response Headers


```
{
  "date": "Sun, 06 May 2018 07:05:46 GMT"
```

www.facebook.com/imcinstitute

IMC Institute Training Courses

June 2018

www.imcinstitute.com

Sming Framework IoT Device (ESP8266) (C++)

4 - 6 June 2018
Instructor: Mr. Sittipong Jansorn

Read More

Big Data Analytics as a Services for Developer

5 - 7 June 2018
Instructor: Assoc. Prof. Dr. Thanachart Numnonda and Mr. Aekanun Thongtae

Read More

Basic Software Testing

12 - 13 June 2018
Instructor: Ms. Cherapa Wannasuk
Special: Registration 3 persons pay only 2 persons

Read More

MicroService on Java Platform

13 - 14 June 2018
Instructor: Mr. Passapong Thaithatgoon

Read More

Test-Driven Development on Java

18 - 19 June 2018
Instructor: Mr. Somkiat Puisungnoen and Mr. Thawatchai Jongsuwanpisan

Read More

Machine Learning for Data Science

20 - 22 June 2018
Instructor: Assoc. Prof. Dr. Thanachart Numnonda and Mr. Aekanun Thongtae

Read More

Introduction to IoT Analytics using Hadoop

26 - 28 June 2018
Instructor: Assoc. Prof. Dr. Thanachart Numnonda and Mr. Aekanun Thongtae

Read More

IMC Institute

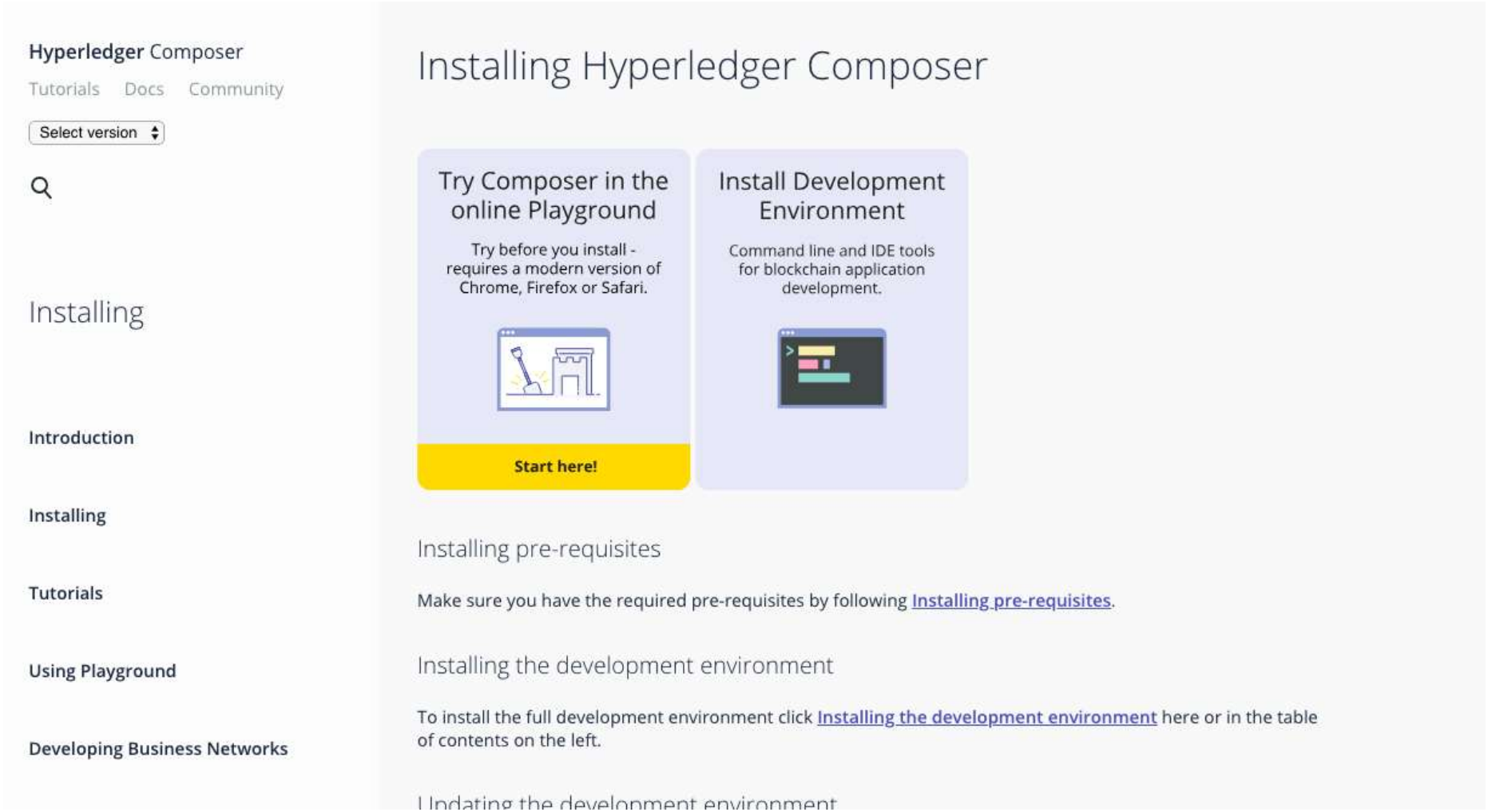
Big Data Analytics

as a Services for Developer

5 - 7 June 2018

www.imcinstitute.com/bigdata-java

Reference:
https://hyperledger.github.io/composer/latest/installing/installing-index.html



Thank you

www.imcinstitute.com
www.facebook.com/imcinstitute