

Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»

Кафедра информатики

Отчет по лабораторной работе #1
« Create a Hyperledger Composer solution»

Карп Александр Игоревич
магистрант кафедры информатики
группа №858641

Проверил: Прудник Александр Михайлович
доцент, кандидат технических наук

Минск 2020

Содержание

1. Установка окружения hyperledger.....	3
2. Создание business network.....	4
3. Rest API.....	7
4. Client service.....	8
5. Выводы.....	9

1. Установка окружения hyperledger

Подготовим среду разработки согласно требованиям и установим необходимые зависимости.

Используем следующее окружение:

- MacOS 10.15.4
- Docker 19.03.8
- Docker Compose 1.13.0
- NodeJS 8.17.0
- npm 6.4.0
- Python 2.7.12

Установим требуемые зависимости:

Essential CLI tools:

```
npm install -g composer-cli@0.20
```

Utility for running a REST Server on your machine to expose your business networks as RESTful APIs:

```
npm install -g composer-rest-server@0.20
```

Useful utility for generating application assets:

```
npm install -g generator-hyperledger-composer@0.20
```

Yeoman is a tool for generating applications, which utilises `generator-hyperledger-composer`:

```
npm install -g yo
```

Установим Hyperledger Fabric:

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

Проверим работоспособность среды:

```
./startFabric.sh  
./createPeerAdminCard.sh
```

Hyperledger Fabric успешно стартовала и admin card сгенерирована со следующими параметрами:

Card name: PeerAdmin@hlfv1
UserId: PeerAdmin

2. Создание Business Network

Создадим бизнес сеть для трансфера денежных средств между пользователями

```
yo hyperledger-composer
```

```
XBS-NB-00045:abric alexanderkarp$ yo hyperledger-composer  
Welcome to the Hyperledger Composer project generator  
? Please select the type of project: Business Network  
You can run this generator using: 'yo hyperledger-composer:businessnetwork'  
Welcome to the business network generator  
? Business network name: wallet  
? Description: wallet  
? Author name: Alexander Karp  
? Author email: karpengold@gmail.com  
? License: Apache-2.0  
? Namespace: org.example.biznet  
? Do you want to generate an empty template network? No: generate a populated sample network
```

Создадим описание модели ассета Transfer, участника User и транзакции Trade в models/org.example.biznet.cto

```
namespace org.example.biznet  
  
asset Transfer identified by transferId {  
  o String transferId  
  o String description  
  o Double amount  
  --> User to  
}  
  
participant User identified by userId {  
  o String userId  
  o String name  
}  
  
transaction Trade {  
  --> Transfer money  
  --> User from  
}
```

Теперь создадим логику обработки транзакции в нашей сети в файле lib/logic.js:

```
/**
```

```

* Track the trade of a commodity from one trader to another
* @param {org.example.mynetwork.Trade} trade - the trade to be processed
* @transaction
*/
async function tradeTransaction(trade) {
  trade.money.to = trade.from;
  let assetRegistry = await getAssetRegistry('org.example.biznet.Transaction');
  await assetRegistry.update(trade.money);
}

```

Пропишем разрешения в сети (все доступно для всех)

```

rule Default {
  description: "Allow all participants access to all resources"
  participant: "ANY"
  operation: ALL
  resource: "org.dmitrysenkovich.personal.*"
  action: ALLOW
}
rule SystemACL {
  description: "System ACL to permit all access"
  participant: "ANY"
  operation: ALL
  resource: "org.hyperledger.composer.system.*"
  action: ALLOW
}

```

Сгенерируем архив hyperledger:

```
composer archive create -t dir -n .
```

Установим нашу сеть:

```
composer network install --card PeerAdmin@hlfv1 --archiveFile wallet@0.0.1.bna
```

Проверим ее работоспособность:

```
composer network start --networkName wallet --networkVersion 0.0.1 --networkAdmin admin --
networkAdminEnrollSecret adminpw --card PeerAdmin@hlfv1 --file networkadmin.card
```

Импортируем карту:

```
composer card import --file networkadmin.card --card admin@wallet
```

3. Rest API

Сгенерируем REST сервер:

```
XBS-NB-00045:wallet alexanderkarp$ composer-rest-server
? Enter the name of the business network card to use: admin@wallet
? 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 the explorer test interface: Yes
? Specify a key if you want to enable dynamic logging:
? Specify if you want to enable event publication over WebSockets: Yes
? Specify if you want to enable TLS security for the REST API: No
```

Посмотрим работоспособность сервера, открыв адрес <http://localhost:3000>



Сервис доступен.

4. Client service

Создадим клиентскую часть для взаимодействия с REST сервером:

```
yo hyperledger-composer:angular
```

```
XBS-NB-00045:wallet alexanderkarp$ yo hyperledger-composer:angular
Welcome to the Hyperledger Composer Angular project generator
? Do you want to connect to a running Business Network? Yes
? Project name: angular-app
? Description: Hyperledger Composer Angular project
? Author name: Alexander Karp
? Author email: karpengold@gmail.com
? License: Apache-2.0
? Name of the Business Network card: admin@wallet
? Do you want to generate a new REST API or connect to an existing REST API? Connect to an existing REST API
? REST server address: http://localhost
? REST server port: 3000
? Should namespaces be used in the generated REST API? Namespaces are not used
Created application!
Completed generation process
create app.js
create Dockerfile
create e2e/app.e2e-spec.ts
create e2e/app.po.ts
create e2e/tsconfig.e2e.json
create e2e/tsconfig.json
create karma.conf.js
create manifest.yml
create package.json
create protractor.conf.js
create proxy.conf.js
create README.md
create src/app/app-routing.module.ts
create src/app/app.component.css
create src/app/app.component.html
create src/app/app.component.spec.ts
create src/app/app.component.ts
create src/app/app.module.ts
create src/app/asset/images/delete_noun_cc.svg
create src/app/asset/images/edit_noun_cc.svg
create src/app/asset/images/failed_noun_cc.svg
create src/app/asset/images/success_noun_cc.svg
create src/app/data.service.ts
create src/app/home/home.component.css
create src/app/home/home.component.html
create src/app/home/home.component.ts
create src/environments/environment.prod.ts
create src/environments/environment.ts
create src/favicon.ico
create src/index.html
create src/main.ts
create src/polyfills.ts
create src/styles.css
create src/test.ts
create src/tsconfig.app.json
create src/tsconfig.json
create src/tsconfig.spec.json
create tsconfig.json
create tslint.json
```

Клиент будет доступен по умолчанию по следующему URL:

<http://localhost:4200>

←

→

↻

localhost:4200/Trade

🔍

☆

🛡️

📄

👤

⋮

wallet@0.0.1.bna

angular-app

Assets

Participants

Transactions

Trade

When invoked, this transaction will have the following attributes:

- money
- from

Invoke

Выводы

В рамках данной работы был изучен процесс установки hyperledger окружения и создания простейшей business network