



Universidade Federal do Rio de Janeiro
Engenharia Eletrônica e de Computação

EEL 880 – Tópicos Especiais em Sistemas Digitais

Hyperledger na Prática: Leilão Utilizando Blockchain

Prof. Otto Carlos Muniz Bandeira Duarte

Aluno: Carlos Felipe Oliveira Leão

Hyperledger: O que é?

“O Hyperledger é um esforço colaborativo “open source” criado para promover as tecnologias de blockchain em vários setores. É uma colaboração global, hospedada pela “The Linux Foundation”, incluindo líderes em finanças, Internet das Coisas, cadeias de suprimento, saúde, manufatura e tecnologia. ”

- Fonte: hyperledger.org

Hyperledger: Objetivos

- Criar estruturas e bases de código de nível corporativo, código-fonte aberto e suporte a transações comerciais;
- Fornecer infraestrutura neutra, aberta e orientada pela comunidade, apoiada pela governança técnica e de negócios;
- Construir comunidades técnicas para desenvolver projetos em blockchain e “ledger” compartilhado, casos de uso e implantações;
- Educar o público sobre as oportunidades de mercado para a tecnologia blockchain;
- Promover a comunidade em comunidades que adotam uma abordagem de kit de ferramentas com muitas plataformas e frameworks.

Hyperledger: Objetivos

- Criar estrutura aberta e segura
 - Fornecer suporte apoiada por
 - Construir e "ledger"
 - Educar o blockchain
 - Promover de kit de ferramentas com muitas plataformas e frameworks.
- Tornar os processos muito mais eficientes, notavelmente mais ágeis, fundamentalmente mais seguros, essencialmente menos burocráticos, substancialmente mais transparentes e com um custo surpreendentemente menor através do blockchain!!!**
- fonte
idade,
blockchain
cnologia
abordagem

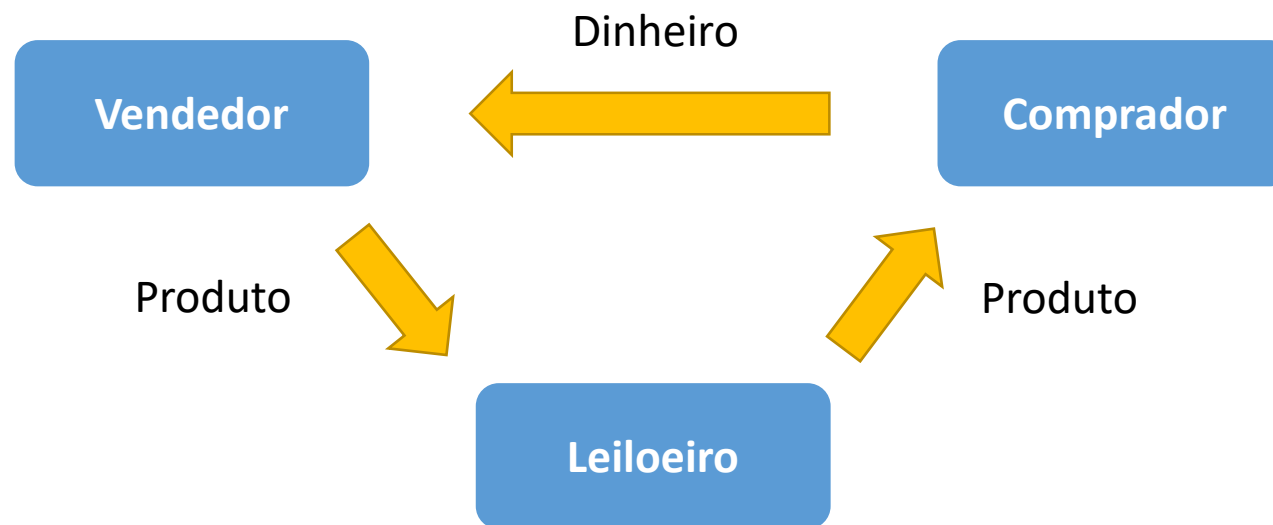
Introdução – Aula Prática

Vamos imaginar agora os participantes de um leilão numa rede privada:

- Vendedor
- Comprador
- Leiloeiro

Introdução – Aula Prática

Tente imaginar também uma rede de transações simplificada em um leilão:



Motivação

- E se os contratos entre os participantes pudessem ter um ID de identificação?
- E se os participantes, sem necessidade de estarem fisicamente no mesmo lugar, pudessem ter acesso às transações ou aos lances do leilão de forma transparente e confiável através do Blockchain?
- E se o pagamento do vencedor do leilão ao vendedor, presente em um contrato inteligente, pudesse ser processado de forma automática em segundos, juntamente com a transferência de propriedade?
- E se as penalidades por atraso na entrega do produto ou não pagamento pelo comprador, presentes no contrato, pudessem ser processadas de forma automática em segundos?
- Mais: e se tivéssemos ao nosso alcance um conjunto de ferramentas que nos permitisse criar essa rede, definir as regras de negócio e fazer os testes de uma forma extremamente simples e prática?

Motivação

- E se os contratos de compra e venda de bens e serviços não fossem feitos no mesmo lugar, poderiam ser feitos de forma transparente?
 - E se o pagamento de um contrato não fosse feito em um prazo determinado?
 - E se as partes envolvidas não fossem obrigadas a pagar pelo comércio eletrônico de forma automática?
- Nosso objetivo, portanto, será fazer essa rede funcionar utilizando um conjunto de ferramentas do Hyperledger para o registro das transações!!**
- Mais: e se tivéssemos ao nosso alcance um conjunto de ferramentas que nos permitisse criar essa rede, definir as regras de negócio e fazer os testes de uma forma extremamente simples e prática?

Qual será a lógica?

- O vendedor criará um produto que será vendido através de um leilão.
- O leilão receberá lances de possíveis compradores.
- Assim que o leilão for fechado o maior lance será validado.
- Por fim será feita a transferência do dinheiro do comprador para o vendedor, os saldos das carteiras serão atualizados e o produto terá um novo dono!

Ferramentas e Pré-Requisitos

Para desenvolvermos este aplicativo em Blockchain utilizaremos o “Hyperledger Composer Playground”.

O “Hyperledger Composer” é um conjunto de ferramentas que torna simples e rápido para proprietários e desenvolvedores de negócios criarem contratos inteligentes e aplicativos Blockchain.

O “Playground” é uma das ferramentas do “Composer” e seu objetivo é acelerar a configuração, a implementação e os testes de uma rede de negócios.

Para utilizar a interface de usuário do Playground precisaremos de um navegador web com acesso à internet.

Configuração de uma Rede de Negócios

Welcome to Hyperledger Composer Playground!



In this web sandbox, you can deploy, edit and test business network definitions. Have a play and learn what Hyperledger Composer Playground is all about.

Let's Blockchain!

? Not sure where to start? [View our Playground tutorial.](#)

Configuração de uma Rede de Negócios


- Acessar o “Playground”: <https://composer-playground.mybluemix.net/login>
- Após isso, clicar em “Let’s Blockchain!” para começarmos a criar a nossa rede!

Configuração de uma Rede de Negócios

Hyperledger Composer Playground

My Business Networks

Connection: Web Browser




Hello, Composer!

Get started with the basic-sample-network, or view our [Playground tutorial](#)

BUSINESS NETWORK

basic-sample-network

Get Started →



Deploy a new business network

Configuração de uma Rede de Negócios

- Clicar em “Deploy a new business network”.
- Ao clicar, irá abrir uma nova tela para configurarmos nossa rede.
- Definir as configurações como nas 2 imagens a seguir e clique em Deploy!

Configuração de uma Rede de Negócios

Deploy New Business Network

1. BASIC INFORMATION

Give your new Business Network a name:

leilao-produtos-network

Describe what your Business Network will be used for:

Esse projeto será um leilão de produtos registrando as transações em um blockchain.

Give the network admin card that will be created a name

admin@leilao-produtos-network

Configuração de uma Rede de Negócios

2. MODEL NETWORK STARTER TEMPLATE

Choose a Business Network Definition to start with:

Choose a sample to play with, start a new project, or import your previous work



basic-sample-
network

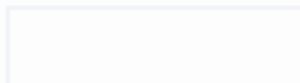
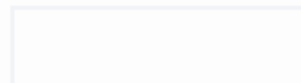


empty-business-
network



Drop here to
upload or [browse](#)

Samples on npm



CONNECTION PROFILE

BASED ON
empty-business-network

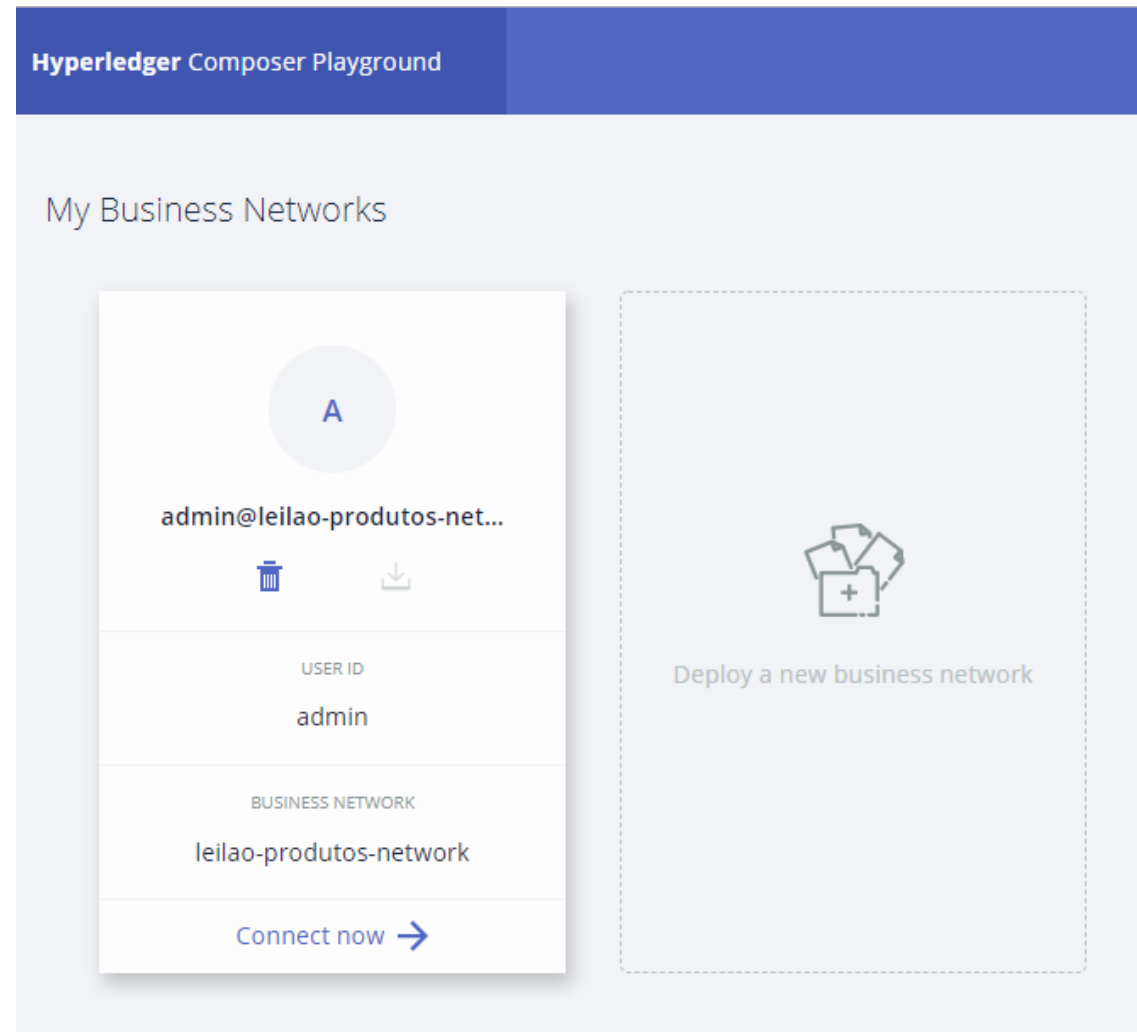
Start from scratch with a blank
business network

Contains: 0 Participant Types, 0
Asset Types, and 0 Transaction
Types

Deploy

Configuração de uma Rede de Negócios

- Rede de negócios criada!
- Agora precisaremos criar nossos participantes, nossos ativos (produto e leilão) e nossas transações (dar um lance e encerrar o leilão).
- Clicar em “Connect now” para começarmos!



Definição dos Parâmetros

Web leilao-produtos-network

Define Test

admin

FILES

About
README.md, package.json

Model File
models/model.cto

Access Control
permissions.acl

Add a file... Export

UPDATE NETWORK

From: 0.0.1

To: 0.0.2-deploy.0

Deploy changes

About File README.md

This is the readme file for the Business Network Definition created in Playground

Ativar o Windows

Acesse Configurações para ativar o Windows.

Legal GitHub

Playground v0.19.12 Tutorial Docs Community

Definição dos Parâmetros

- Como podemos ver, temos de início 3 arquivos em nossa rede:
 - About -> arquivo “README”, normalmente explicativo sobre o projeto.
 - Access Control -> contém as regras de controle de acesso de cada usuário. Como em nossa rede todos os usuários devem ter acesso ao valor das transações, este arquivo não será alterado.
 - Model File -> arquivo em que definiremos o modelo de nossa rede de negócios. Usaremos a linguagem de modelagem CTO.

Linguagem de Modelagem (CTO)

- Como falamos anteriormente, precisaremos definir nossos participantes, ativos e transações.
- Para defini-los, usaremos a linguagem de modelagem padrão do Hyperledger Composer chamada de CTO (uma homenagem ao nome original do projeto, “Concerto”).
- Essa linguagem é focada na modelagem da rede de negócios com algumas palavras-chave, sendo assim muito simples e intuitiva (principalmente para quem já trabalhou com linguagens orientadas a objeto).
- Caso se interesse em saber mais sobre a linguagem, acesse:
https://hyperledger.github.io/composer/latest/reference/cto_language



Definição dos Parâmetros

- Voltando a nossa definição de parâmetros após uma pequena pausa para falar sobre a linguagem CTO, vamos criar nosso arquivo “model.cto”.
- Deve-se clicar no arquivo “Model File” do “Playground” e colar nele o código do arquivo “.txt” “Model-File.txt”. Este passo serve para abreviarmos nossa
- Veja que é neste momento em que definimos os participantes, ativos e transações de nossa rede de negócios.

Definição dos Parâmetros

Web leilao-produtos-network

Define Test

admin

FILES

About
README.md, package.json

Model File
models/org.example.model.cto

Script File
lib/script.js

Access Control
permissions.acl

Add a file...

Export

UPDATE NETWORK

From: 0.0.1

To: 0.0.2-deploy.0

Deploy changes

Script File lib/script.js

```
1  /**
2   * Este arquivo tem o objetivo de definir a regra de negócio
3   * para as transações do leilão deste exemplo.
4   * @param {org.example.model.Lance} lance - Transação como o Lance que será adicionado ao Leilão
5   * @transaction
6   */
7
8  function criarLance(lance) {
9      var leilao = lance.leilao;
10     if (leilao.estadoLeilao !== 'aberto') {
11         throw new Error('O leilão ainda não está aberto!');
12     }
13
14     if (lance.comprador.saldo < lance.valor) {
15         throw new Error('O comprador não possui saldo suficiente para efetuar a transação!');
16     }
17
18     if (lance.comprador.membroId === leilao.produto.proprietario.membroId) {
```

Everything looks good!

Any problems detected in your code would be reported here

Ativar o Windows

Acesse Configurações para ativar o Windows.

Playground v0.19.12 Tutorial Docs Community



Definição dos Parâmetros

- Após definirmos o modelo de nossa rede, precisaremos definir agora nossa regra de negócios, ou seja, as regras de como irá funcionar o nosso leilão.
- Para isso, precisaremos clicar em “Add a file” e escolher a opção “Script File (.js)”. Como a extensão do arquivo já diz, devemos criar um código em Javascript que diga como irá funcionar nossa rede de negócios (quem poderá comprar o produto, o dono do maior lance é quem ganhará, etc.)
- Agora, vamos copiar o código do arquivo “.txt” chamado de “Script-File-JS” e colar na tela do “Playground”.
- Por último devemos clicar em “Deploy changes” para finalizarmos a definição dos parâmetros!

Teste da Rede de Negócios

- Após a configuração da nossa rede e a definição dos parâmetros, chegou a hora de testar criando nosso primeiro leilão!
- Clicar em “Test”, situado no “header” (parte de cima da tela).
- O primeiro passo será criar os participantes. Veja a próxima imagem:

Teste da Rede de Negócios

Web leilao-produtos-network

DefineTest

admin

PARTICIPANTS

Membro

ASSETS

Leilao

Produto


TRANSACTIONS

All Transactions

Submit Transaction

Participant registry for org.example.model.Membro

+ Create New Participant

ID	Data
	
<div><div>This registry is empty!</div><div>To create resources in this registry click create new at the top of this page</div></div>	

Ativar o Windows

Acesse Configurações para ativar o Windows.

Playground v0.19.12

Tutorial

Docs

Community

Legal

GitHub

Criação dos Participantes





- Clicar em “Membro” e depois em “+ Create New Participant”
- Após isso, criar 3 participantes.
- Usar como exemplo os dados das 2 próximas imagens.



Criação dos Participantes

```
1  {  
2    "$class": "org.example.model.Membro",  
3    "membroId": "E1",  
4    "nome": "Fundão Airlines",  
5    "cnpj": "11.111.111/0001-00",  
6    "saldo": 100000  
7  }
```

Criação dos Participantes

ID	Data	
E1	<pre>{ "\$class": "org.example.model.Membro", "membroId": "E1", "nome": "Fundão Airlines", "cnpj": "11.111.111/0001-00", "saldo": 100000 }</pre>	 
	<div>Collapse</div>	
E2	<pre>{ "\$class": "org.example.model.Membro", "membroId": "E2", "nome": "Brasil Airlines", "cnpj": "00.000.000/0001-00", "saldo": 2000000 }</pre>	 
	<div>Collapse</div>	
E3	<pre>{ "\$class": "org.example.model.Membro", "membroId": "E3", "nome": "China Airlines", "cnpj": "55.555.555/0001-55", "saldo": 9000000 }</pre>	 
	<div>Collapse</div>	

Criação do Produto

- Clicar em “Produto” e depois em “+ Create New Asset”
- Após isso, definir as características do produto como na tela ao lado.

Create New Asset

In registry: **org.example.model.Produto**

JSON Data Preview

```
1  {  
2    "$class": "org.example.model.Produto",  
3    "produtoId": "aviao1",  
4    "nome": "Boeing 777",  
5    "descricao": "Aviao seminovo muito bem conservado.",  
6    "valor": 1500000,  
7    "proprietario": "resource:org.example.model.Membro#E1"  
8  }
```

☐ Optional Properties

Just need quick test data? [Generate Random Data](#)

Cancel

Create New

Criação do Leilão

- Clicar em “Leilão” e depois em “+ Create New Asset”
- Após isso, definir as características do leilão como na tela ao lado.
- Clicar em “All Transactions” para ver os registros das transações!

Create New Asset

In registry: **org.example.model.Leilao**

JSON Data Preview

```
1  {  
2    "$class": "org.example.model.Leilao",  
3    "leilaoId": "Leilao1",  
4    "estadoLeilao": "aberto",  
5    "valorInicial": 1000000,  
6    "valorFinal": 2000000,  
7    "produto": "resource:org.example.model.Produto#aviao1"  
8  }
```

☐ Optional Properties

Just need quick test data? [Generate Random Data](#)

Cancel

Create New

Veja a Blockchain Funcionando!

Web leilao-produtos-network		Define	Test	admin	
PARTICIPANTS					
Membro					
ASSETS					
Leilao					
Produto					
TRANSACTIONS					
All Transactions					
Submit Transaction					

Oferta do 1º Lance pelo Produto

- Vamos agora clicar em “Submit Transaction”.
- Em “Transaction Type” escolher “Lance” e definir um lance como na figura ao lado:

Submit Transaction

Transaction Type Lance

JSON Data Preview

```
1 {  
2   "$class": "org.example.model.Lance",  
3   "valor": 1300000,  
4   "leilao": "resource:org.example.model.Leilao#Leilao1",  
5   "comprador": "resource:org.example.model.Membro#E2"  
6 }
```

☐ Optional Properties

Just need quick test data? [Generate Random Data](#)

Cancel Submit

Oferta do 2º Lance pelo Produto

- Agora vamos dar um 2º lance pelo produto, afinal temos que ter uma disputa!
- Clicar novamente em “Submit Transaction”, escolher “Lance” e definir os parâmetros como na figura ao lado:

Submit Transaction

Transaction Type Lance

JSON Data Preview

```
1 {  
2   "$class": "org.example.model.Lance",  
3   "valor": 2000000,  
4   "leilao": "resource:org.example.model.Leilao#Leilao1",  
5   "comprador": "resource:org.example.model.Membro#E3"  
6 }
```

☐ Optional Properties

Just need quick test data? [Generate Random Data](#) Cancel Submit

Encerramento do Leilão

- Para encerrar o leilão e definir quem vai ficar com o produto, clicar novamente em “Submit Transaction”.
- Definir apenas o id do Leilão como na figura ao lado.

Submit Transaction

Transaction Type

EncerrarLeilao

JSON Data Preview

```
1 {  
2   "$class": "org.example.model.EncerrarLeilao",  
3   "leilao": "resource:org.example.model.Leilao#Leilao1"  
4 }
```

☐ Optional Properties

Just need quick test data? [Generate Random Data](#)

Cancel

Submit

Será que funcionou?

- Para validar que realmente as transações ocorreram, precisamos primeiro verificar se houve o encerramento do leilão.
- Após isso, vamos verificar se houve a transferência do maior lance dado, da empresa “E3” para a empresa “E1”, e a atualização do saldo da carteira das 2.
- Por último, vamos verificar se o produto possui um novo dono (E3 – China Airlines) registrado!

Validação das Transações

PARTICIPANTS

Membro

ASSETS

Leilao

Produto

TRANSACTIONS

All Transactions

Submit Transaction







Asset registry for org.example.model.Leilao

+ Create New Asset

ID	Data
Leilao1	<div><pre>{ "\$class": "org.example.model.Leilao", "leilaoId": "Leilao1", "estadoLeilao": "fechado", "valorInicial": 1000000, "valorFinal": 2000000, "lances": [{ "\$class": "org.example.model.lance", "valor": 1300000, "leilao": "resource:org.example.model.Leilao#Leilao1", "comprador": "resource:org.example.model.Membro#E2", "transactionId": "80564e9e-7e90-46b7-b742-a78351232e1e", "timestamp": "2018-07-17T21:11:04.065Z" }, { "\$class": "org.example.model.lance", "valor": 2000000, "leilao": "resource:org.example.model.Leilao#Leilao1", "comprador": "resource:org.example.model.Membro#E3", "transactionId": "52e92538-a953-4281-8754-f9f2838787f9", "timestamp": "2018-07-17T21:13:10.076Z" }], "produto": "resource:org.example.model.Produto#aviao1" }</pre></div> <div>Collapse</div>



Validação das Transações

ID	Data	
E1	<pre>{ "\$class": "org.example.model.Membro", "membroId": "E1", "nome": "Fundão Airlines", "cnpj": "11.111.111/0001-00", "saldo": 2100000 }</pre>	 
	<button>Collapse</button>	
E2	<pre>{ "\$class": "org.example.model.Membro", "membroId": "E2", "nome": "Brasil Airlines", "cnpj": "00.000.000/0001-00", "saldo": 2000000 }</pre>	 
	<button>Collapse</button>	
E3	<pre>{ "\$class": "org.example.model.Membro", "membroId": "E3", "nome": "China Airlines", "cnpj": "55.555.555/0001-55", "saldo": 7000000 }</pre>	 
	<button>Collapse</button>	

Validação das Transações

PARTICIPANTS

Membro

ASSETS

Leilao

Produto



TRANSACTIONS

All Transactions

Submit Transaction

Asset registry for org.example.model.Produto

+ Create New Asset

ID	Data	
aviao1	<pre>{ "\$class": "org.example.model.Produto", "produtoId": "aviao1", "nome": "Boeing 777", "descricao": "Aviao seminovo muito bem conservado.", "valor": 2000000, "proprietario": "resource:org.example.model.Membro#E3" }</pre>	 

Collapse

Conclusão

- Após analisar as 3 imagens anteriores, vimos que o leilão foi encerrado, os saldos do vendedor e do comprador foram atualizados e o produto possui um novo dono. Ou seja, missão cumprida!
- Percebemos, portanto, que de uma forma rápida e dinâmica não só podemos criar aplicativos utilizando Blockchain, como também podemos testá-los e validá-los, graças ao conjunto de ferramentas do “Hyperledger Composer”.
- Ou seja, foi criada uma rede de negócios com a transparência e segurança do Blockchain de forma extremamente simples!
- Agora que você já viu na prática como fazer uma aplicação em Blockchain funcionar, não se limite a apenas este exemplo. Use sua criatividade e mãos a obra!