



# CheckPoint

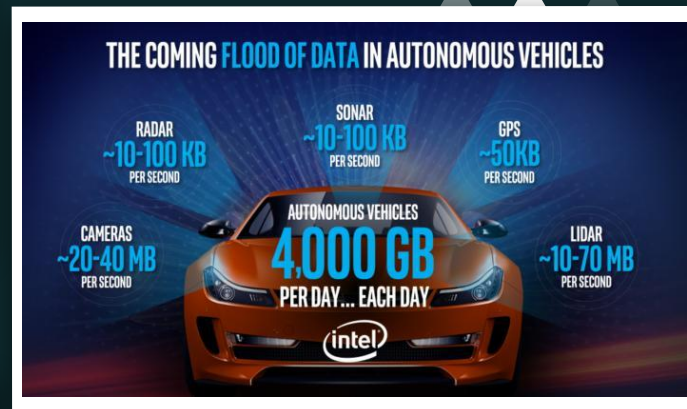
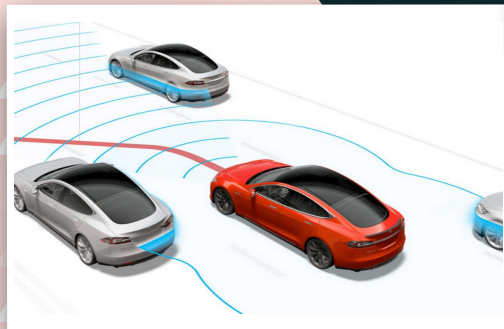
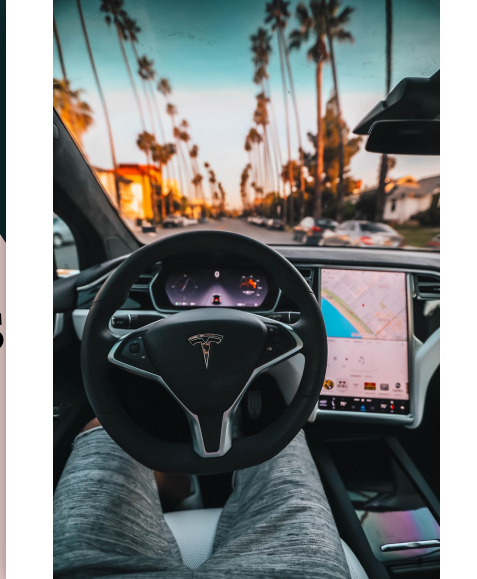
Prof. Jó Ueyama

Sistemas Computacionais Distribuídos e suas  
técnicas de implementação

Bruner Eduardo  
William Alves


# SSCo904 – Sistemas Computacionais Distribuídos

Tesla's data  
collection



# Empecillo

## Curva de Aprendizagem



Contents

- [Documentation](#)
- [Downloads](#)
- [References](#)
- [Developer](#)
- [Extensions](#)

### Simulation of Urban MOBility

**SUMO User Conference 2020**  
The SUMO User Conference 2020 [✉](#) takes place **May, 11-14, 2020** in Berlin.

"Simulation of **Urban MOBility**" (Eclipse SUMO) is an open source, highly portable, microscopic and continuous road traffic simulation package designed to handle large road networks. SUMO is [licensed](#) under the [Eclipse Public License V2](#) [✉](#). "Eclipse SUMO" is a trademark of the Eclipse Foundation. If you use SUMO, please support its development by telling us about your [Publications](#).

The content of this Documentation is freely editable according to the [wiki style](#) [✉](#). That means, whenever You find a solution to a problem mentioned on the [mailing lists](#), feel free to add an article to this Documentation or an entry to the [frequently asked questions](#). Write access is restricted. In order to edit this Documentation, click on the "Edit on GitHub" button on the upper right corner and submit a Pull Request. Here is a short help on [editing articles](#).

This Documentation is continuously updated and always refers to the latest development version. Documentation for a specific release version of SUMO is included in the download of that version.

### Documentation

- [Frequently Asked Questions \(FAQ\)](#)
- [SUMO User Documentation](#)
- [Screenshots](#)
- [Installing SUMO](#)
- [Tutorials](#)
- [Contact Information](#)
- [ChangeLog](#)

### Downloads



# Supply Chain

Prof. Jó Ueyama

Sistemas Computacionais Distribuídos e suas  
técnicas de implementação



# Índice

## Até o momento

A Blockchain

API e Middlewares

*Smart Contracts*

Paradigmas distribuídos

Motivação

Próximos passos

## Fluxo do checkpoint

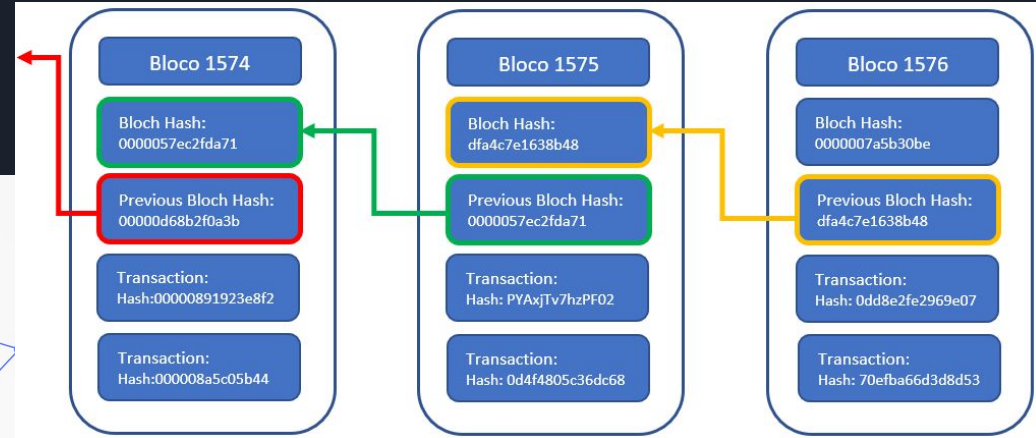
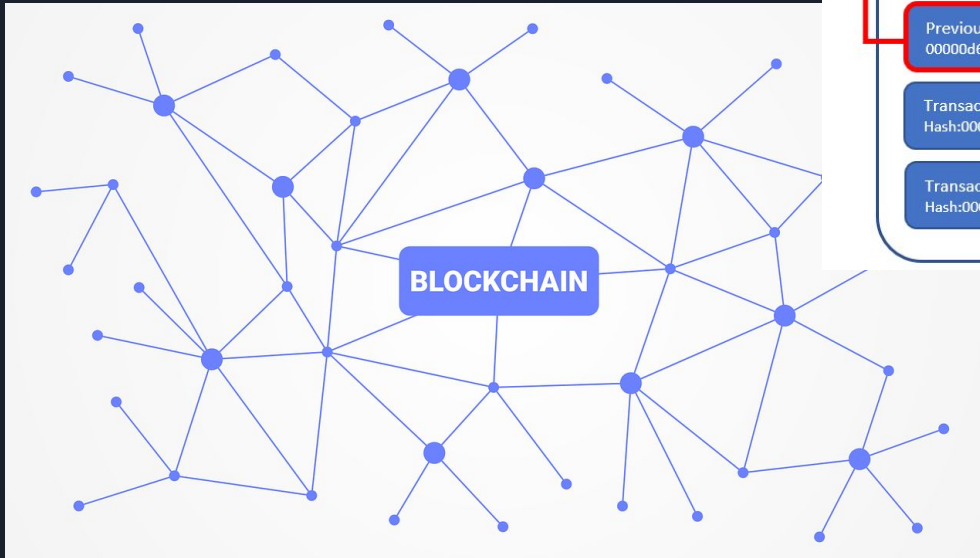
- ❑ Conceitos de blockchain
- ❑ Aplicações de blockchain para a disciplinas de Sistemas Computacionais Distribuídos
- ❑ Passos do Projeto



# Report



# Conceito Base





## What is blockchain?

Let's talk about  
the shared ledger  
technology poised to  
revolutionize the way  
the world works.



Why do we need blockchain?

[@illiamw/Blockchain/blob/master/blockchain.pdf](https://github.com/illiamw/Blockchain/blob/master/blockchain.pdf)




# Block





```
JS block.js  ×  JS chain.js  🔍
BlockchainSimplificada > JS block.js > ...


  9  export function generateHash({index, prevHash, timestamp, data})
10  |   return SHA256(data + index + prevHash + timestamp).toString();
11  | }
12
13  export function create(data) {
14  |   const lastBlock = chain.last();
15  |
16  |   const newblock = {
17  |     index : lastBlock.index + 1,
18  |     prevHash : lastBlock.hash,
19  |     timestamp : new Date().getTime(),
20  |     data : data,
21  |   }
22  |   newblock.hash = this.generateHash(newblock);
23  |   return newblock;
24  | }
25
26  export function validateBlock(newBlock, lastBlock = chain.last())
```

# Uma breve demonstração

 illiamw / Blockchain

 illiamw Update README.md

 BlockchainSimplificada	Update block.js
 .gitattributes	Initial commit
 README.md	Update README.md
 blockchain.pdf	Infografico blockchain IBM

 README.md

## Blockchain

Para avaliação de aplicação da disciplina Sistemas Computacionais Distribuídos Paulo, apresenta-se uma proposta de aplicação

### Demonstração de um rede blockchain

# Pilares

# Criptografia

```
/**
 * @var crypto Variavel Global com os metodos de criptografia.
 */
const crypto = require('..../node_modules/crypto-js');
const chain = require('..../chain').default;
/**
 * @module generateHash geração de hash criptografado
 */
module.exports.generateHash = function({index, prevHash, timestamp, data}) {
  return crypto.SHA256(data + index + prevHash + timestamp).toString();
}
```

# Pilares

# Comunicação

## O SOCKET.IO 2.0 ESTÁ AQUI

APRESENTANDO O MOTOR EM TEMPO REAL MAIS RÁPIDO E CONFIÁVEL

```
~ / Projetos / tweets / index.js  
1. var io = require ('socket.io') (80) ;  
2. var cfg = require ('./ config.json');  
3. var tw = require ('nó-tweet-stream')  
   (cfg);  
4. tw.track ('socket.io');  
5. tw.track ('javascript');  
6. tw.on ('tweet', função (tweet) {  
7.   io.emit ('tweet', tweet) ;  
8. });
```





# Pilares

# Orquestração Server

```
app.post('/transaction', (req, res) => {
  const { sender, receiver, amount } = req.body;
  io.emit(SocketActions.ADD_TRANSACTION, sender, receiver, amount);
  res.json({ message: 'transaction success' }).end();
});

app.get('/chain', (req, res) => {
  res.json(blockChain.toArray()).end();
});

io.on('connection', (socket) => {
  console.info(`Socket connected, ID: ${socket.id}`);
  socket.on('disconnect', () => {
    console.log(`Socket disconnected, ID: ${socket.id}`);
  });
});

blockChain.addNode(socketListeners(client(`http://localhost:${PORT}`), blockChain));

httpServer.listen(PORT, () => console.info(`Express server running on ${PORT}...`));
```



## Pilares

## Validação

```
module.exports.validateBlock = (newBlock, lastBlock = chain.last()) => {  
  let blockIsValid = false;  
  
  if (newBlock.index == lastBlock.index+1) {  
    blockIsValid = true;  
  } else if (newBlock.prevHash == lastBlock.hash) {  
    blockIsValid = true;  
  } else if (newBlock.hash == this.generateHash(newBlock)) {  
    blockIsValid = true  
  }  
  
  return blockIsValid;  
}
```





API



Plataforma de gerenciamentos de rede blockchain.

Visualização de propagação.



**HYPERLEDGER**  
**FABRIC**

Estrutura da rede e contratos inteligentes.

Parceria com a IBM.

Código aberto.

 **THE LINUX FOUNDATION** PROJECTS

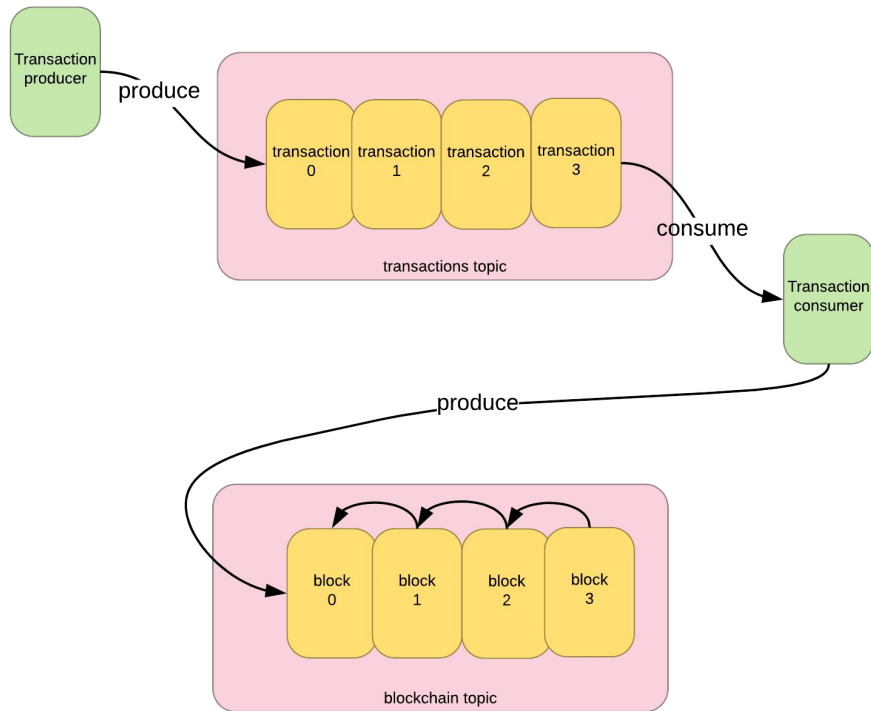
# API




Ordenação

Imutabilidade

Orientado a evento



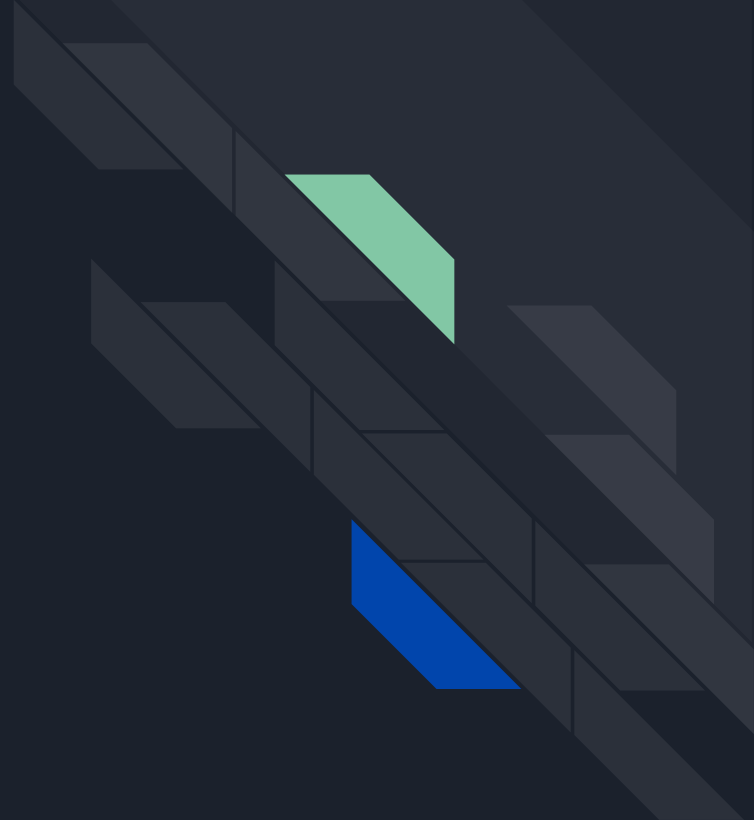



“Blockchain é uma  
solução esperando por  
problemas para resolver”.

Gerald Nash



# Motivação

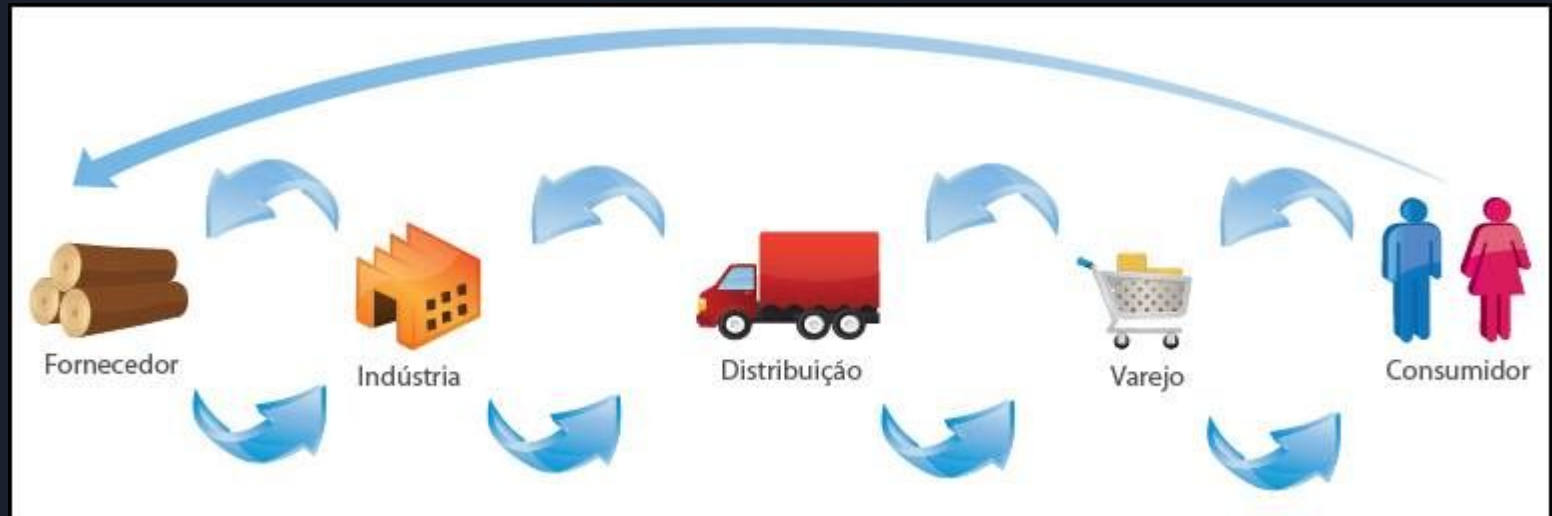




Supply Chain é um  
mercado de US\$  
64 trilhões

Abril, 2019

# Cadeia de suprimentos





# Domínio: Validação com foco na origem



# Domínio: Validação com foco na origem

 Conjunto de dados

 Grupos

 Fluxo de Atividades

## Sistema Nacional de Informações Florestais - SNIF

O Serviço Florestal Brasileiro tem como uma de suas competências, estabelecidas na Lei nº 11.284, de 2 de março de 2006, criar e manter o Sistema Nacional de Informações Florestais - SNIF, integrado ao Sistema Nacional de Informações sobre o Meio Ambiente - SINIMA.

O SNIF tem como objetivo geral coleccionar e produzir, organizar, armazenar, processar e disseminar dados, informações e conhecimentos sobre as florestas e o setor florestal, de modo a subsidiar projetos e políticas que conciliem o uso e a conservação das florestas do Brasil.

Amazônia

Amazônia Legal

Biomás

Certificado florestal

Certificação Florestal

Desmatamento

Extração de Produto...

Floresta

Floresta plantada

Focos de Calor

Incêndios Florestais

Informações florestais

Produtos florestais...

Produtos florestais...

SFB

SNIF

Serviço Florestal B...

Terras Indígenas

UCs

Unidades de Conserv...

Estes dados estão disponíveis como o esperado?

4

 Sim

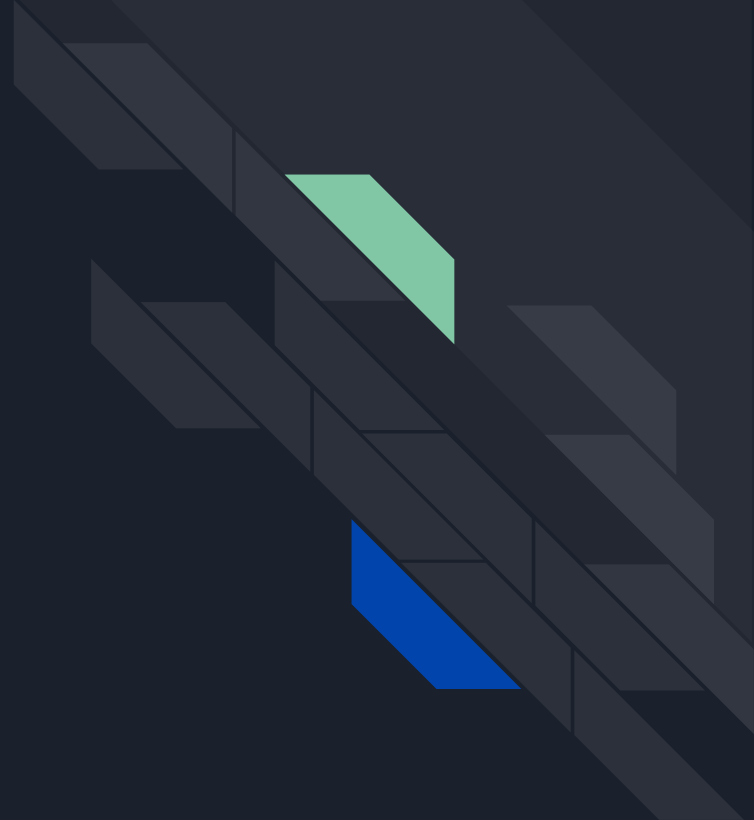
ou

 Não

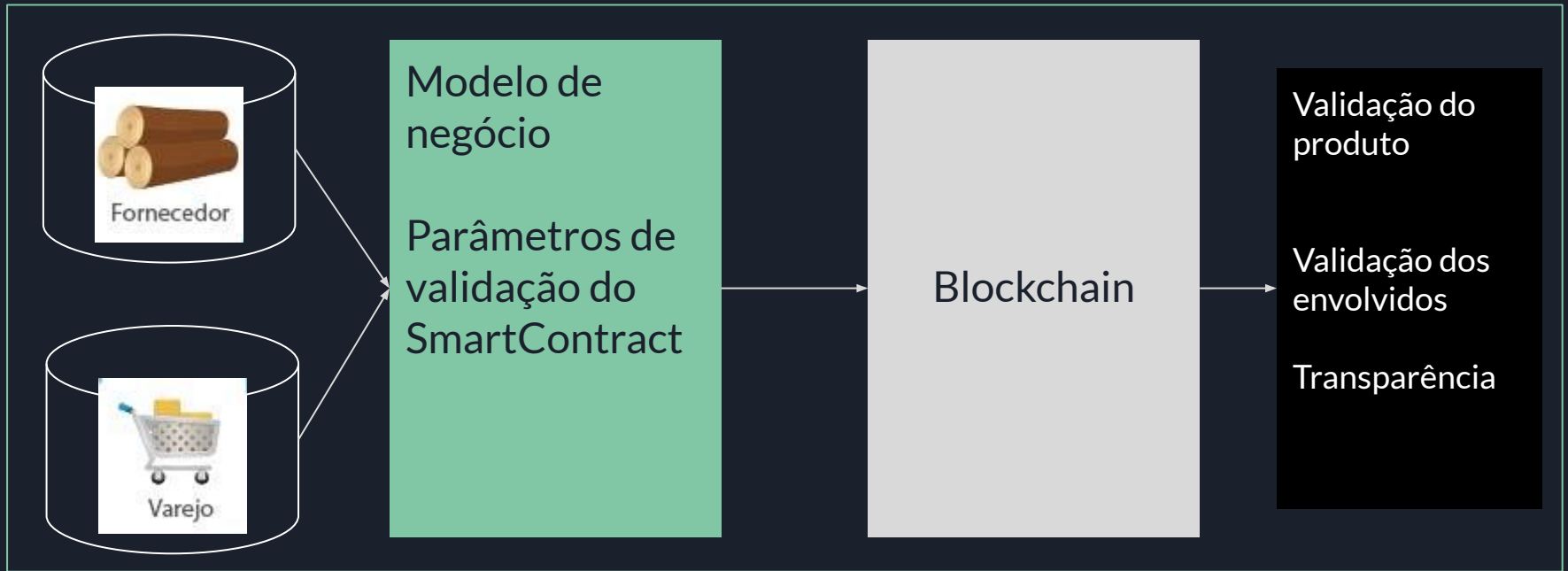
12



Próximos passos



# Simular





Obrigado !

Repositório Git



**illiamw / Blockchain**