

Desafio de Projeto - Bootcamp Blockchain Developer

Crie o seu NFT de POKEMON com Blockchain

Projeto destina-se a desenvolver competências para criar contratos inteligentes. Mais especificamente para este desafio, criar um contrato no padrão ERC721, um token NFT, e simular um jogo de batalhas entre pokemons.

Consiste basicamente das seguintes atividades:

- Preparação e ativação do ambiente;
- Implementação do token ERC721;
- Publicar o contrato na Blockchain;
- Realizar batalha com os pokémons;

Tecnologias envolvidas

- Solidity – Linguagem de programação (<https://soliditylang.org/>)

Linguagem de programação.

- Ganache (<https://archive.trufflesuite.com/ganache/>)

Ambiente de rede local - Blockchain

- MetaMask (<https://metamask.io/download/>)

Gerenciador de carteiras digitais

- Remix IDE (<https://remix.ethereum.org/>)

Interface para desenvolvimento, deploy e testes do smart contract.

- IPFS (<https://docs.ipfs.tech/install/ipfs-desktop/>)

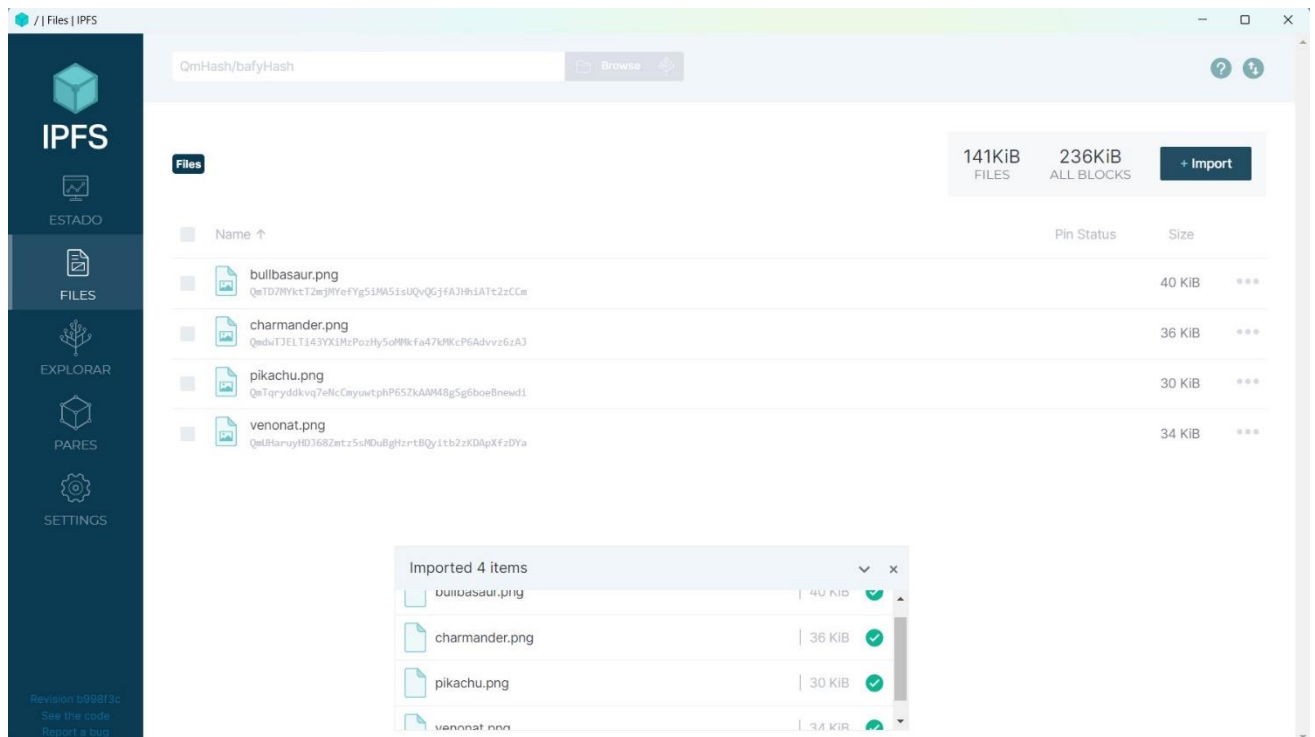
Sistema de gerenciamento de arquivos em blockchain

- Openzeppelin (<https://www.openzeppelin.com/>)

Provedor de bibliotecas de padrões de smart contracts.

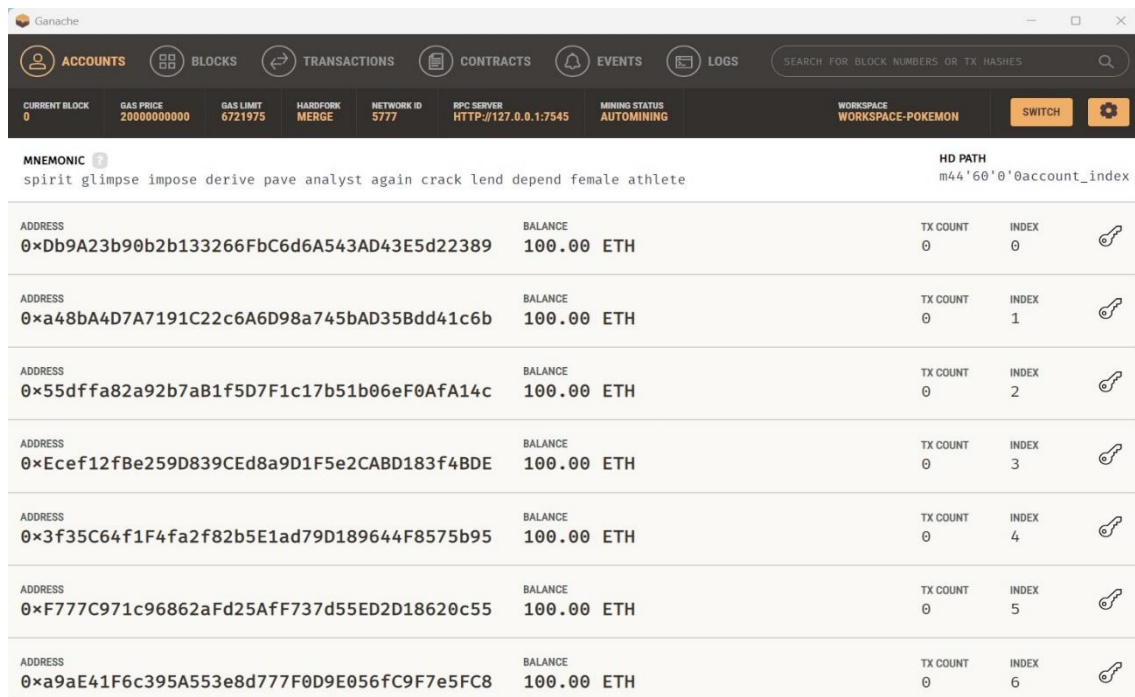
O passo a passo para concluir o desafio

1º - Obter imagens (ícones) de pokemons e carregá-las no IPFS



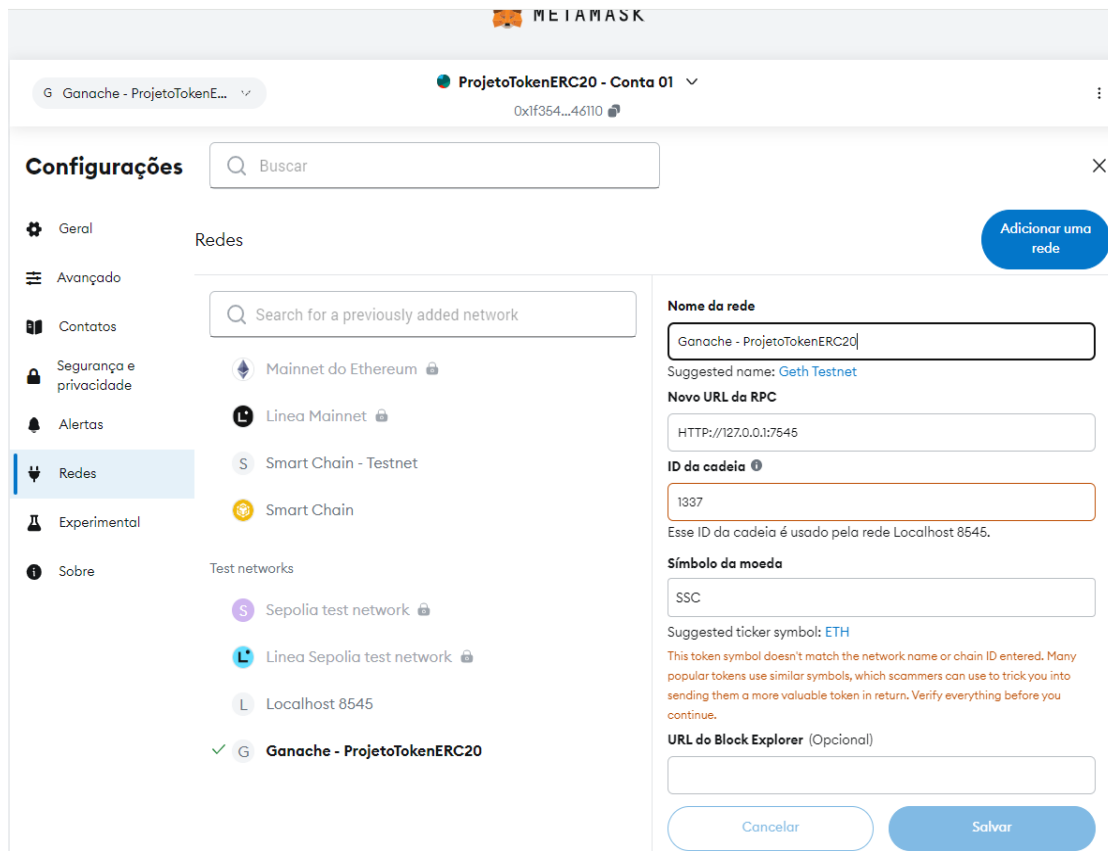
Imagens obtidas de site de ícones gratuitos abaixo para os quatro pokemons listados acima. <https://www.flaticon.com/br/icones-gratis/pokemon>

2º - Ativação do ambiente Ganache

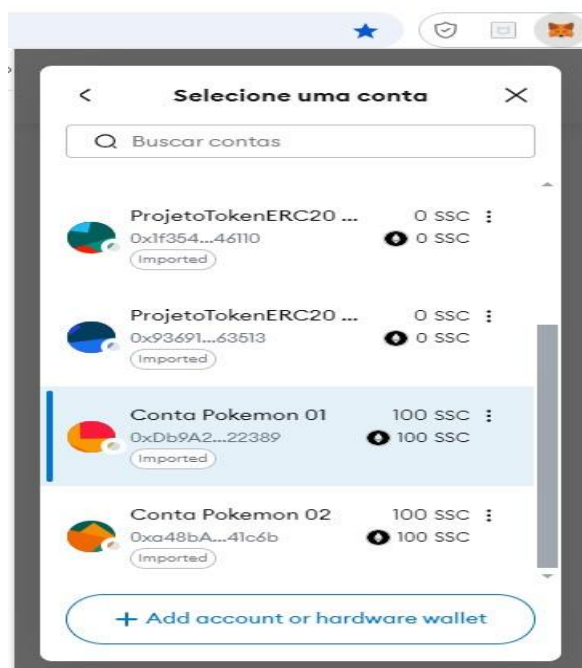


Criado o workspace WORKSPACE-POKEMON para este desafio, onde foram adicionados automaticamente 10 endereços de carteiras para a operação na rede blockchain.

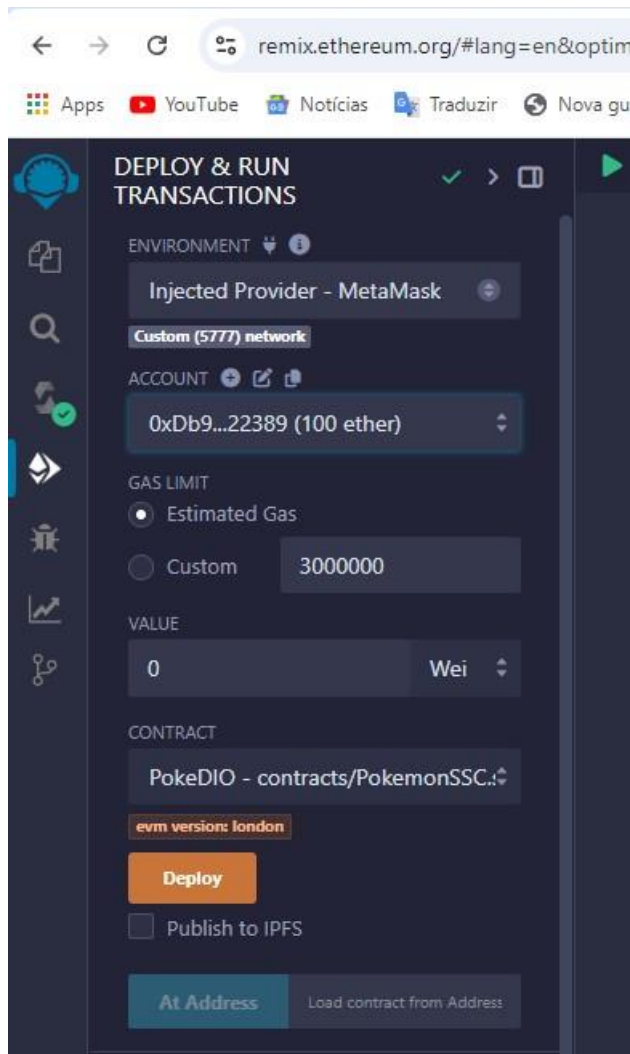
3º - Configuração de contas na MetaMask no google chrome



Dado que o MetaMask já fora instalado e a rede ganache já havia sido adicionada em exercícios anteriores, por ora, fez-se a importação de duas novas cotas referentes aos dois primeiros endereços do workspace criado no ganache, como ilustrado abaixo. Contas **Conta Pokemon 01** e **Conta Pokemon 02**.



4º - Ativação da RemixIDE e conexão com a blockchain ganache por meio da MetaMask



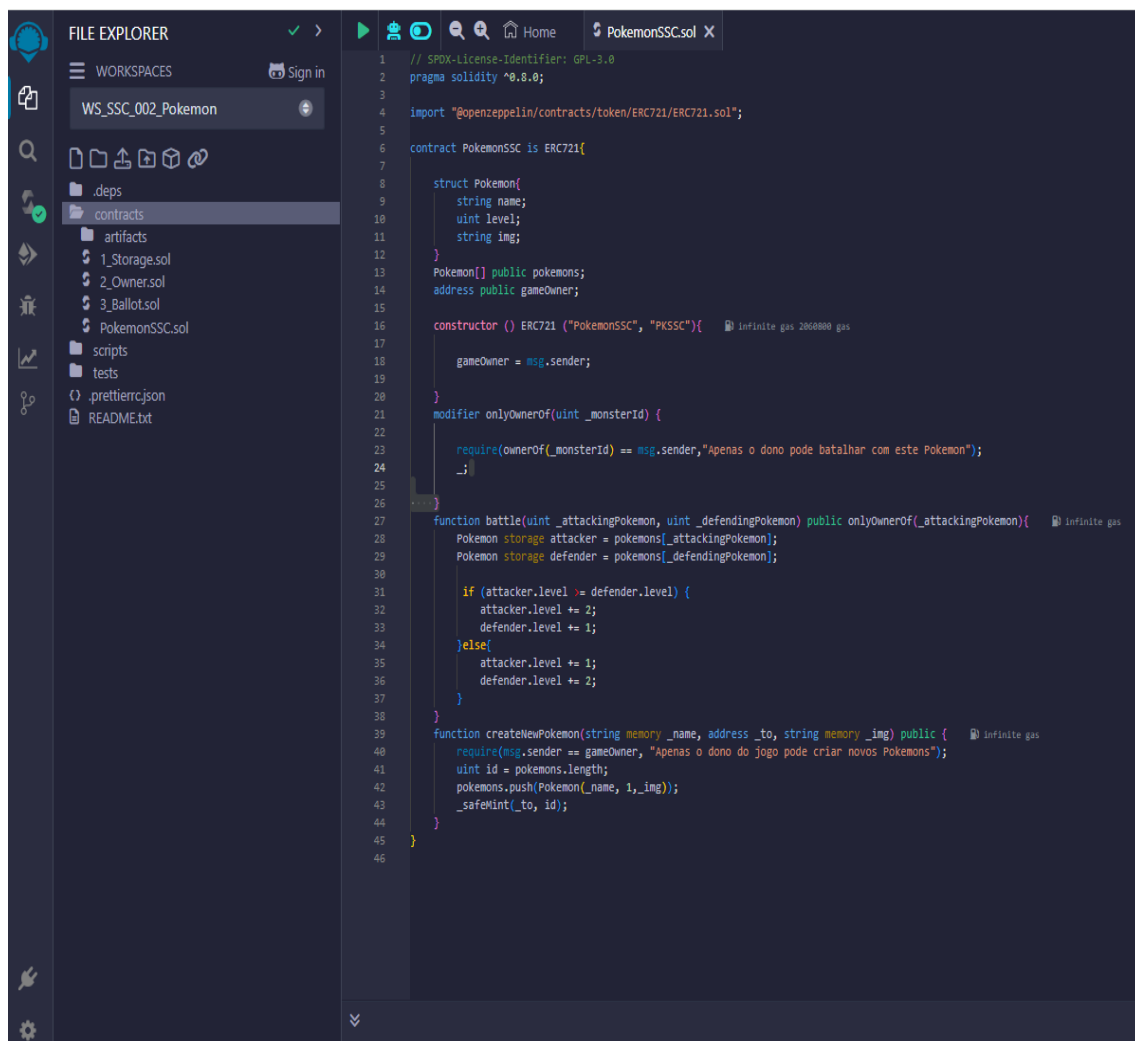
RemixIDE ativada e conectada à Blockchain por meio do Injected Provider – MetaMask, tendo sido escolhida a Conta Pokemon 01, endereço final 22389, para ser a conta owner do contrato.

5º - Implementação do Smart Contract ERC751 – NFT -Pokemon

A implementação foi feita com a importação do código disponibilizado pelo Prof. Cassiano, no Github do desafio. A única alteração feita foi no nome do contrato para PokemonSSC.

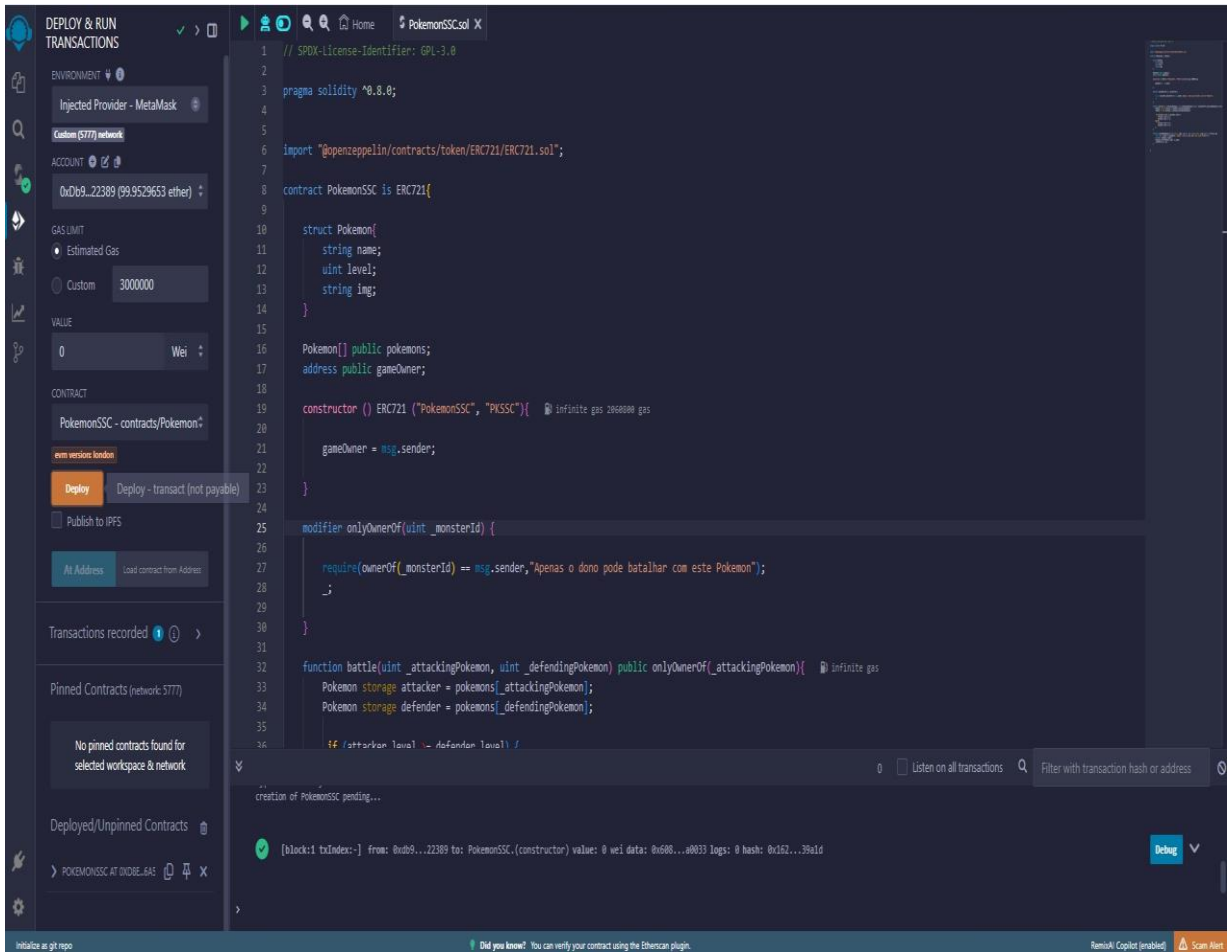
A programação do contrato foi feita em novo workspace, WS_SSC_002_Pokemon, no arquivo PokemonSSC.sol.

Segue código do contrato.

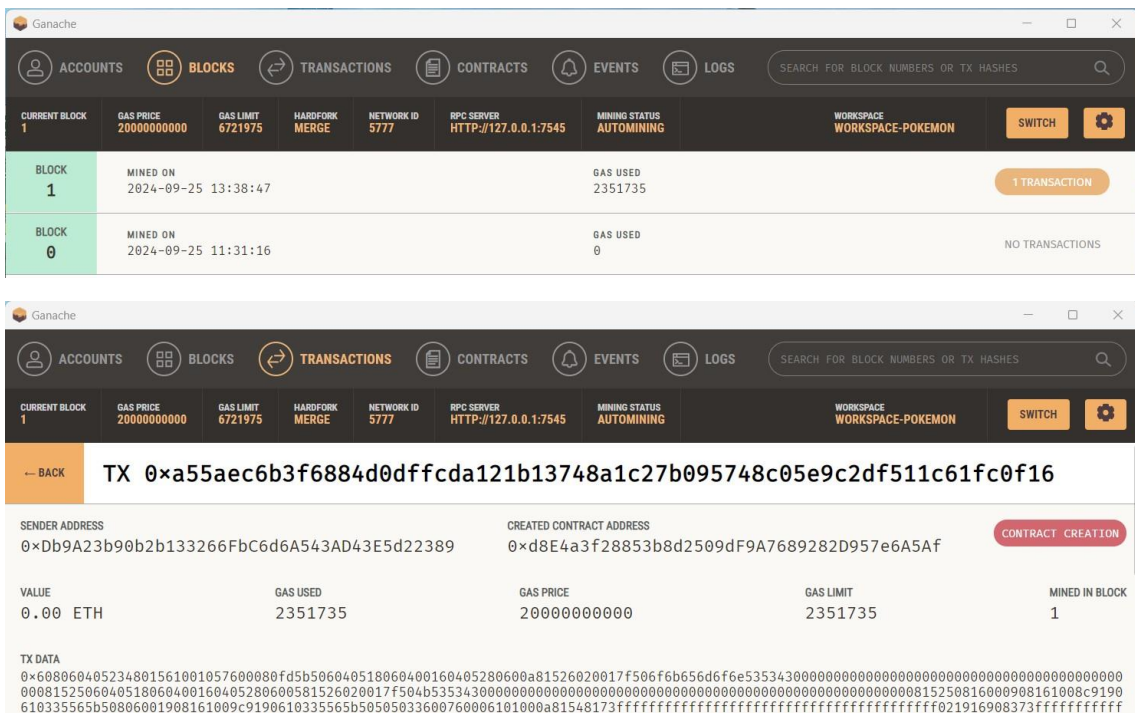


```
1 // SPDX-License-Identifier: GPL-3.0
2 pragma solidity ^0.8.0;
3
4 import "@openzeppelin/contracts/token/ERC721/ERC721.sol";
5
6 contract PokemonSSC is ERC721{
7
8     struct Pokemon{
9         string name;
10        uint level;
11        string img;
12    }
13    Pokemon[] public pokemons;
14    address public gameOwner;
15
16    constructor () ERC721 ("PokemonSSC", "PKSSC"){
17        gameOwner = msg.sender;
18    }
19
20    modifier onlyOwnerOf(uint _monsterId) {
21        require(ownerOf(_monsterId) == msg.sender, "Apenas o dono pode batalhar com este Pokemon");
22    }
23
24
25
26
27    function battle(uint _attackingPokemon, uint _defendingPokemon) public onlyOwnerOf(_attackingPokemon){
28        Pokemon storage attacker = pokemons[_attackingPokemon];
29        Pokemon storage defender = pokemons[_defendingPokemon];
30
31        if (attacker.level >= defender.level) {
32            attacker.level += 2;
33            defender.level += 1;
34        }else{
35            attacker.level += 1;
36            defender.level += 2;
37        }
38    }
39
40    function createNewPokemon(string memory _name, address _to, string memory _img) public {
41        require(msg.sender == gameOwner, "Apenas o dono do jogo pode criar novos Pokemons");
42        uint id = pokemons.length;
43        pokemons.push(Pokemon(_name, 1, _img));
44        _safeMint(_to, id);
45    }
46 }
```

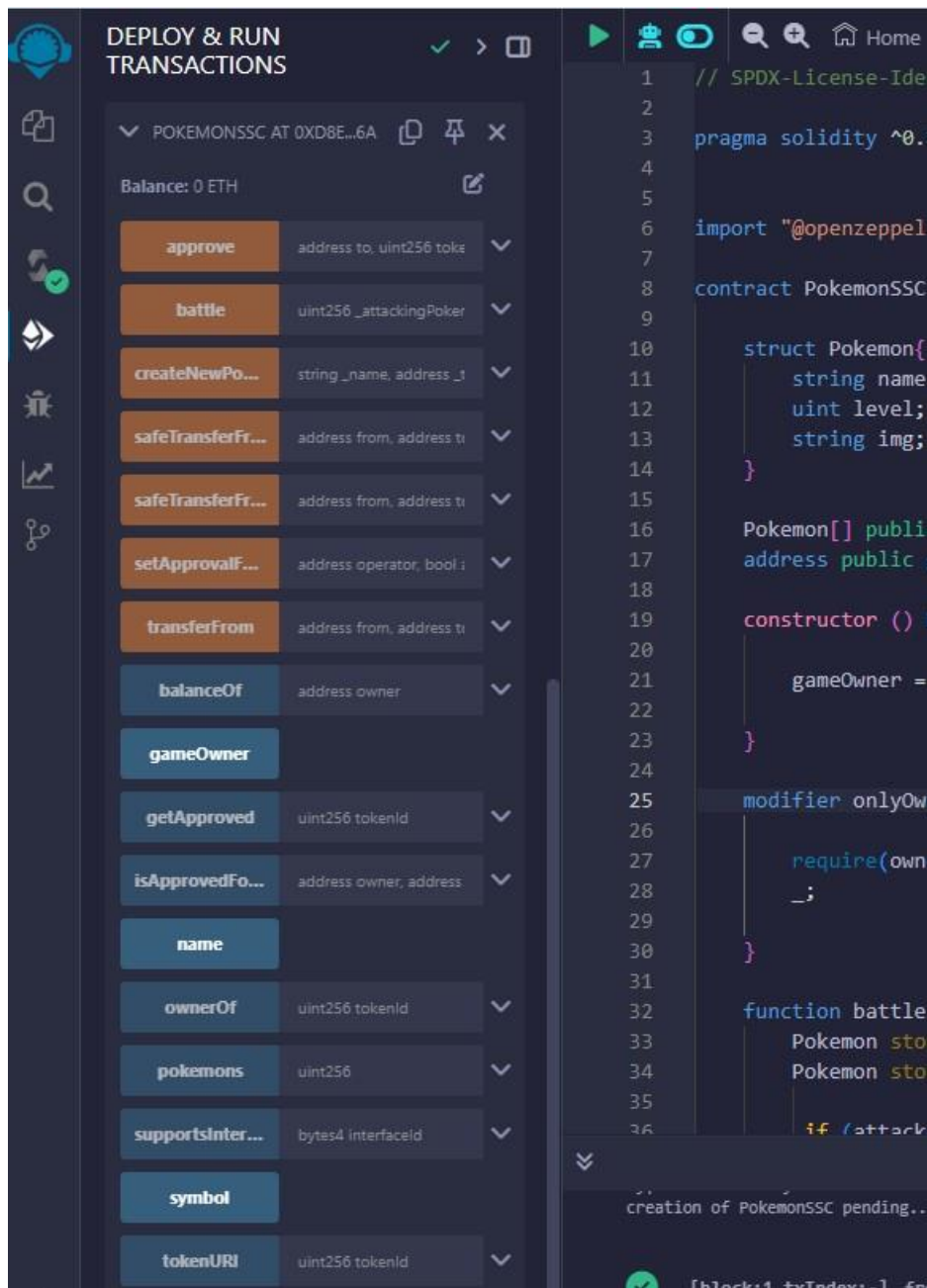
6º - Compilação e Deploy do Contrato PokemonSSC



Verificando o bloco e transação de criação do contrato no Ganache.



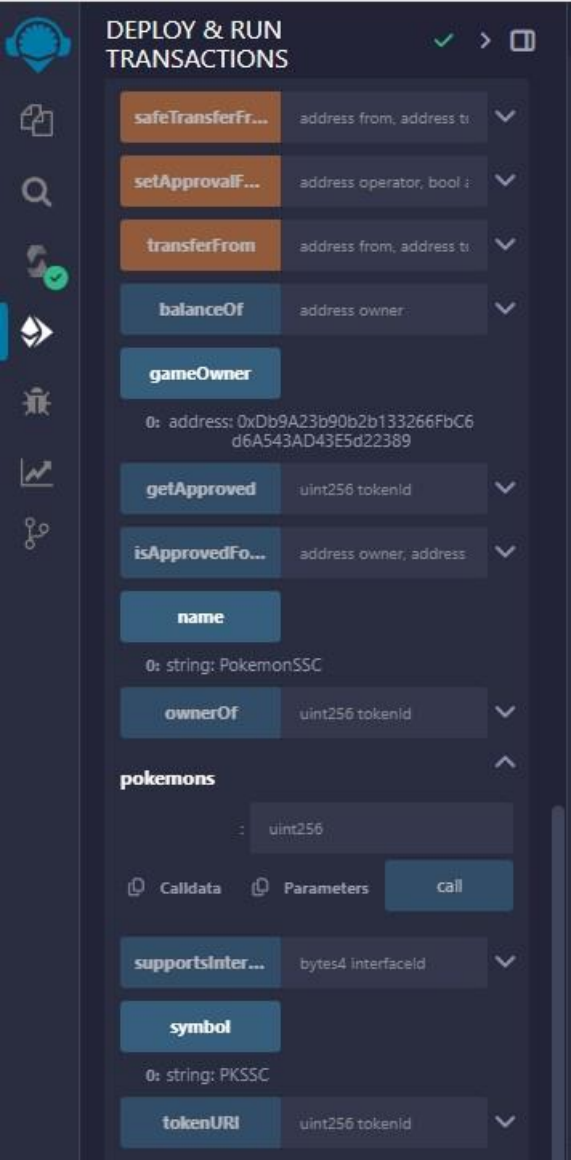
7º - Consultando a interface do contrato no RemixIDE



The screenshot displays the Remix IDE interface. On the left, the 'DEPLOY & RUN TRANSACTIONS' panel shows the 'POKEMONSSC' contract at address 0XD8E...6A. The panel lists various methods and properties of the contract, including 'approve', 'battle', 'createNewPo...', 'safeTransferFr...', 'setApprovalF...', 'transferFrom', 'balanceOf', 'gameOwner', 'getApproved', 'isApprovedFo...', 'name', 'ownerOf', 'pokemons', 'supportsInter...', 'symbol', and 'tokenURI'. The right panel shows the Solidity code for the 'PokemonSSC' contract. The code includes a struct for 'Pokemon' with fields 'name', 'level', and 'img'. It also includes a public array 'pokemons' of type 'Pokemon[]', a constructor, and a 'battle' function. The bottom status bar indicates 'creation of PokemonSSC pending..'

Observa-se que boa quantidade de métodos e propriedades do contrato foram herdados pela inclusão da biblioteca de padrão ERC721 obtida da Openzeppelin.

Na figura a seguir pode-se observar o estado inicial do contrato. Entre as informações pode-se constatar o endereço do owner do contrato, o nome (PokemonSSC) e símbolo (PKSSC) do contrato.



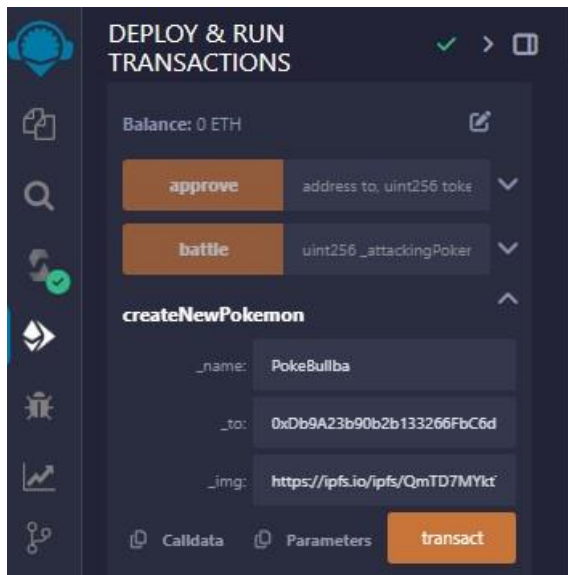
8º - Criando novos pokemons

Foram criados dois pokémons: PokeBullba e PokePika conforme segue.

PokeBullba:

_to: <endereço da conta 01 – owner>

_img: <link do arquivo contendo o ícone do Buslbasaur no IPFS>



Ao transacionar, é solicitada a confirmação por meio da conta MetaMask ativa, como em todas as transações.

Da mesma forma foi criado o pokémon PokePika.

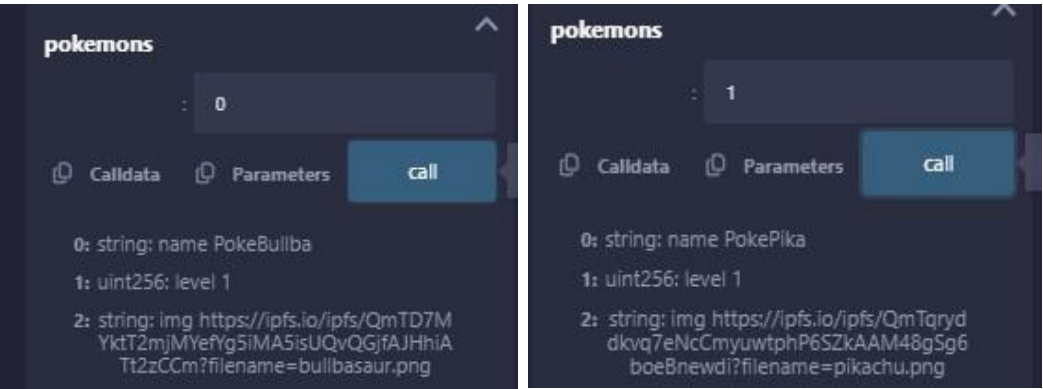
PokePika:

_to: <endereço da conta 02 >

_img: <link do arquivo contendo o ícone do Pikachu no IPFS>

.

9º - Consultado os pokemons criados com a RemixIDE



Observe-se que ambos estão no nível 1, que deverá variar conforme batalharem.

10º - Consultando blocos e transações da criação de pokemons

Ganache

ACCOUNTS

BLOCKS

TRANSACTIONS

CONTRACTS

EVENTS

LOGS

SEARCH FOR BLOCK NUMBERS OR TX HASHES

CURRENT BLOCK3

GAS PRICE20000000000

GAS LIMIT6721975

HARDFORKMERGE

NETWORK ID5777

RPC SERVERHTTP://127.0.0.1:7545

MINING STATUSAUTOMINING

WORKSPACEWORKSPACE-POKEMON

SWITCH

BLOCK3

MINED ON2024-09-25 16:33:46

GAS USED217015

1 TRANSACTION

BLOCK2

MINED ON2024-09-25 16:25:09

GAS USED231675

1 TRANSACTION

BLOCK1

MINED ON2024-09-25 13:38:47

GAS USED2351735

1 TRANSACTION

BLOCK0

MINED ON2024-09-25 11:31:16

GAS USED0

NO TRANSACTIONS

Ganache

ACCOUNTS

BLOCKS

TRANSACTIONS

CONTRACTS

EVENTS

LOGS

SEARCH FOR BLOCK NUMBERS OR TX HASHES

CURRENT BLOCK3

GAS PRICE20000000000

GAS LIMIT6721975

HARDFORKMERGE

NETWORK ID5777

RPC SERVERHTTP://127.0.0.1:7545

MINING STATUSAUTOMINING

WORKSPACEWORKSPACE-POKEMON

SWITCH

TX HASH

0x92596baf332ff5051af85d0e4a0aecee2ef1d5966e2c7d6c1cca2afd5cce1c29

CONTRACT CALL

FROM ADDRESS0xDB9A23b90b2b133266FbC6d6A543AD43E5d22389

TO CONTRACT ADDRESS0xd8E4a3f28853b8d2509dF9A7689282D957e6A5Af

GAS USED217015

VALUE0

TX HASH

0xc847a7726fd082adbcd7d887f3b18abe4b48dbb7ff00089ace3fa64cff5b796cd

CONTRACT CALL

FROM ADDRESS0xDB9A23b90b2b133266FbC6d6A543AD43E5d22389

TO CONTRACT ADDRESS0xd8E4a3f28853b8d2509dF9A7689282D957e6A5Af

GAS USED231675

VALUE0

TX HASH

0xa55aec6b3f6884d0dffcd121b13748a1c27b095748c05e9c2df511c61fc0f16

CONTRACT CREATION

FROM ADDRESS0xDB9A23b90b2b133266FbC6d6A543AD43E5d22389

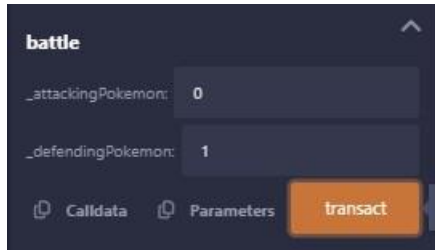
CREATED CONTRACT ADDRESS0xd8E4a3f28853b8d2509dF9A7689282D957e6A5Af

GAS USED2351735

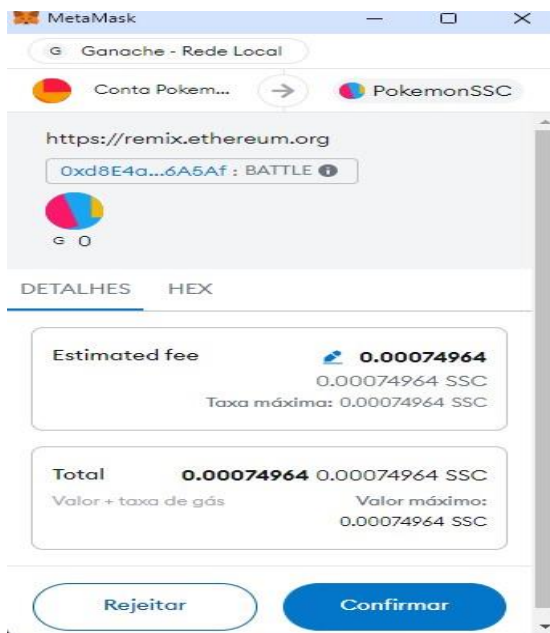
VALUE0

11º - Colocando PokeBullba e PokePika para batalhar

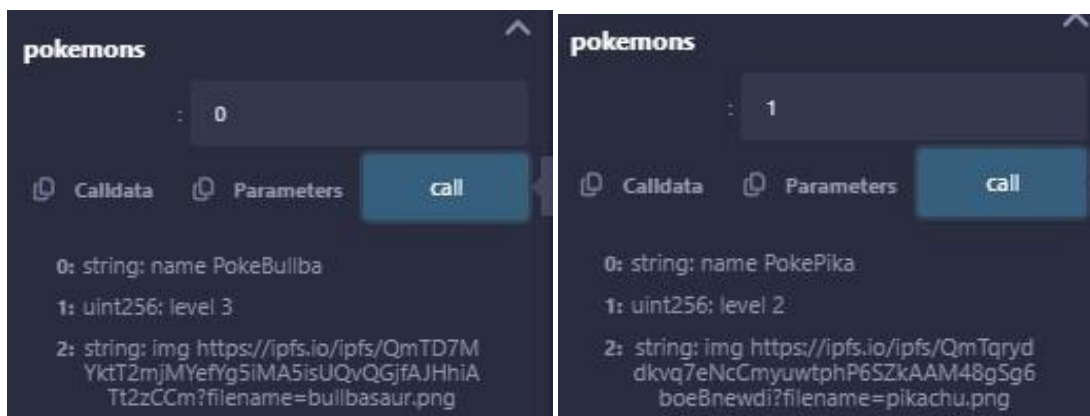
Nesta batalha, o PokeBullba atacará p PokePika



MetaMask solicita conformação para transacionar

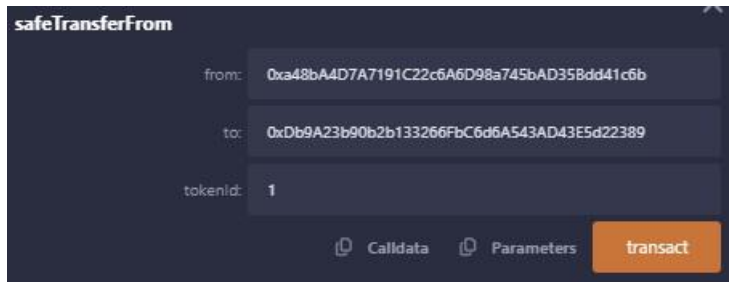


Consultando pokemons após a batalha

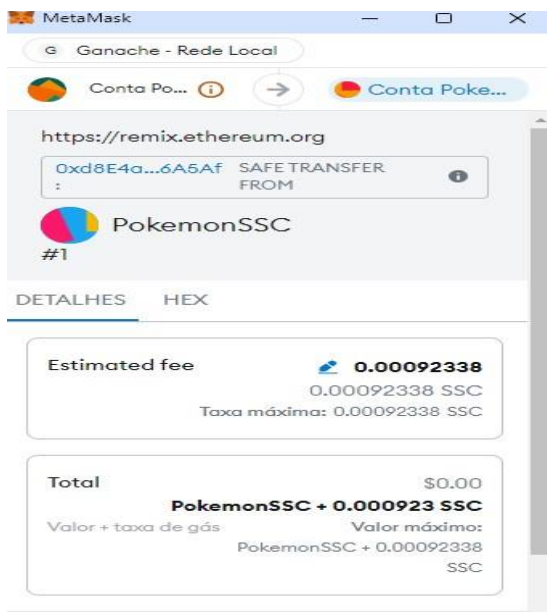


Observa-se a alteração de níveis como resultado da batalha.

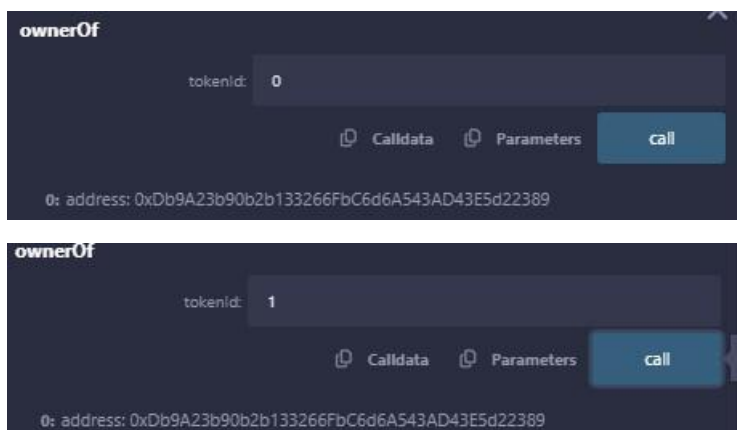
12º - Transferindo o PokePika para a conta owner



MetaMask solicita confirmação



Consultando o proprietário de ambos os pokemons após a transferência



Observa-se o endereço do owner igual, o que conforma o sucesso da transferência.

Consultando blocos e transações após a transferência

ACCOUNTS

BLOCKS

TRANSACTIONS

CONTRACTS

EVENTS

LOGS

SEARCH FOR BLOCK NUMBERS OR TX HASHES

CURRENT BLOCK5

GAS PRICE2000000000

GAS LIMIT6721975

HARDFORKMERGE

NETWORK ID5777

RPC SERVERHTTP://127.0.0.1:7545

MINING STATUSAUTOMINING

WORKSPACEWORKSPACE-POKEMON

SWITCH

BLOCK5	MINED ON 2024-09-25 17:23:56	GAS USED 41369	1 TRANSACTION
BLOCK4	MINED ON 2024-09-25 17:02:07	GAS USED 37482	1 TRANSACTION
BLOCK3	MINED ON 2024-09-25 16:33:46	GAS USED 217015	1 TRANSACTION
BLOCK2	MINED ON 2024-09-25 16:25:09	GAS USED 231675	1 TRANSACTION
BLOCK1	MINED ON 2024-09-25 13:38:47	GAS USED 2351735	1 TRANSACTION
BLOCK0	MINED ON 2024-09-25 11:31:16	GAS USED 0	NO TRANSACTIONS

ACCOUNTS

BLOCKS

TRANSACTIONS

CONTRACTS

EVENTS

LOGS

SEARCH FOR BLOCK NUMBERS OR TX HASHES

CURRENT BLOCK5

GAS PRICE2000000000

GAS LIMIT6721975

HARDFORKMERGE

NETWORK ID5777

RPC SERVERHTTP://127.0.0.1:7545

MINING STATUSAUTOMINING

WORKSPACEWORKSPACE-POKEMON

SWITCH

TX HASH 0xf302fd67bd3bb4230b00375ffc5e2a3006ba2fc5dce19287f3418ddf0f417589	CONTRACT CALL		
FROM ADDRESS 0xa48bA4D7A7191C22c6A6D98a745bAD35Bdd41c6b	TO CONTRACT ADDRESS 0xd8E4a3f28853b8d2509dF9A7689282D957e6A5Af	GAS USED 41369	VALUE 0
TX HASH 0xa0c30150247cd07dbc121c7e3d57c8be24f9061a3b3a99d7a8af9c62bd6f9874	CONTRACT CALL		
FROM ADDRESS 0xDb9A23b90b2b133266FbC6d6A543AD43E5d22389	TO CONTRACT ADDRESS 0xd8E4a3f28853b8d2509dF9A7689282D957e6A5Af	GAS USED 37482	VALUE 0
TX HASH 0x92596baf332ff5051af85d0e4a0aecee2ef1d5966e2c7d6c1cca2afd5cce1c29	CONTRACT CALL		
FROM ADDRESS 0xDb9A23b90b2b133266FbC6d6A543AD43E5d22389	TO CONTRACT ADDRESS 0xd8E4a3f28853b8d2509dF9A7689282D957e6A5Af	GAS USED 217015	VALUE 0
TX HASH 0xc847a7726fd082adbc7d887f3b18abe4b48dbb7ff00089ace3fa64cff5b796cd	CONTRACT CALL		
FROM ADDRESS 0xDb9A23b90b2b133266FbC6d6A543AD43E5d22389	TO CONTRACT ADDRESS 0xd8E4a3f28853b8d2509dF9A7689282D957e6A5Af	GAS USED 231675	VALUE 0
TX HASH 0xa55aec6b3f6884d0dffcd121b13748a1c27b095748c05e9c2df511c61fc0f16	CONTRACT CREATION		

Sobre mim

Eu sou Sérgio Santa Catarina, servidor público aposentado após longa carreira em TI, começando como programador em ambientes mainframe IBM, linguagem COBOL/CICS, e passando por diversas outras tecnologias e funções/cargos, como analista de suporte, administrador de banco de dados, analista de negócios e, por fim, gerente de projetos. Antes de me aposentar ainda trabalhei 6 anos com gestão de processos e planejamento estratégico. Se tem uma linguagem que ainda me atrevo a codificar algo é SQL.

Hoje busco me atualizar nas tecnologias atuais com ênfase em blockchain e demais que dão suporte às criptomoedas. Não faço ideia se vou trabalhar com isso. No momento busco aprender.

Conecte-se comigo

Perfil DIO; https://www.dio.me/users/sergio_santacatarina)

E-mail: (<mailto:sergio.santacatarina@gmail.com>)

LinkedIn: (www.linkedin.com/in/sérgio-santa-catarina-95b58273/)

[Instagram (<https://www.instagram.com/sergiosantacatarina/>)

Apêndice

Apenas lembrando que sou iniciante nestas tecnologias.

Autores

- [[@SergioSantaCatarina](https://www.github.com/SergioSantaCatarina)](<https://www.github.com/SergioSantaCatarina>)