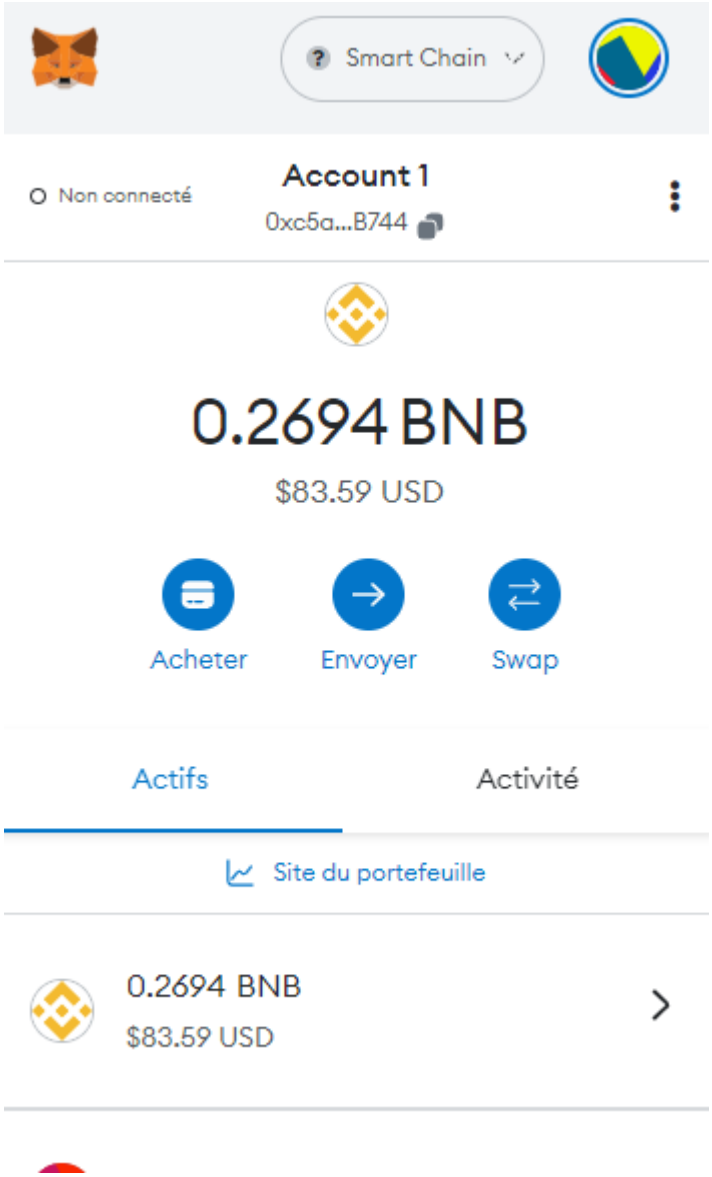
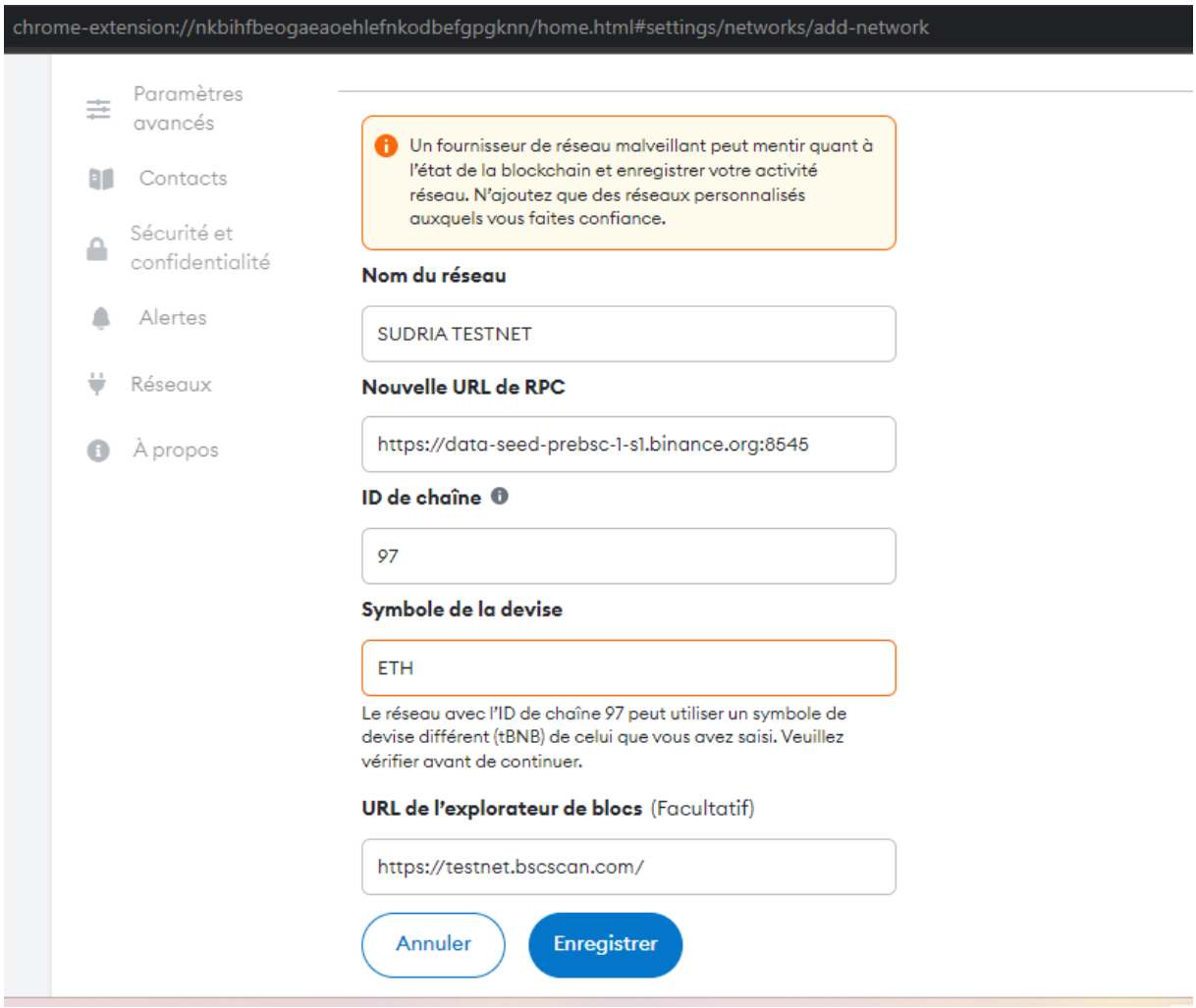
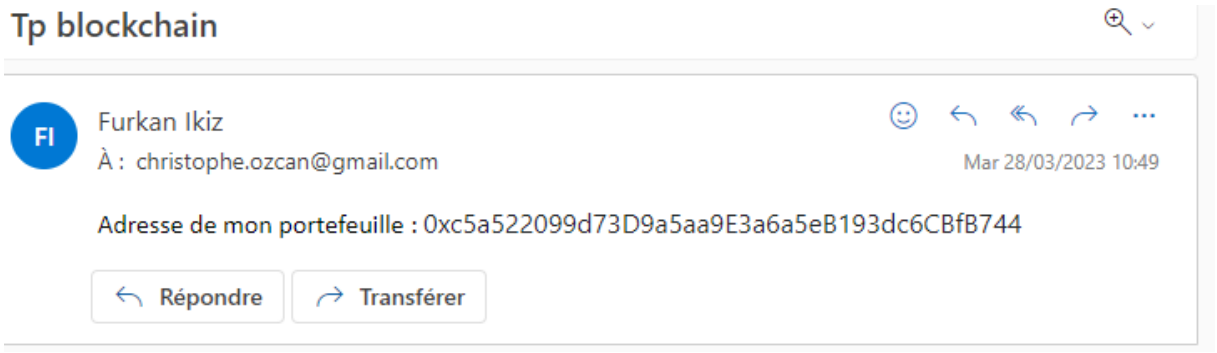
a)b)c) J’avais déjà un compte Metamask :



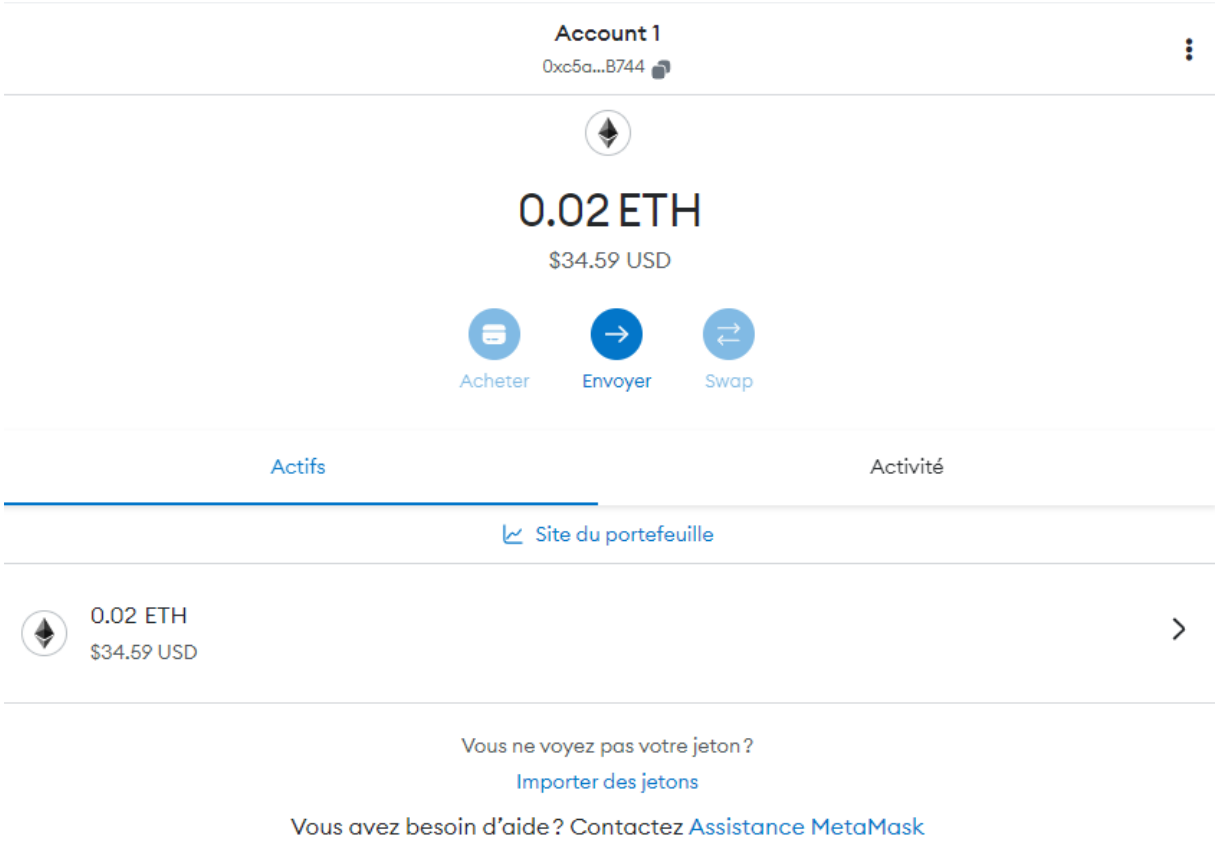
d) J’ai ajouté un réseau :



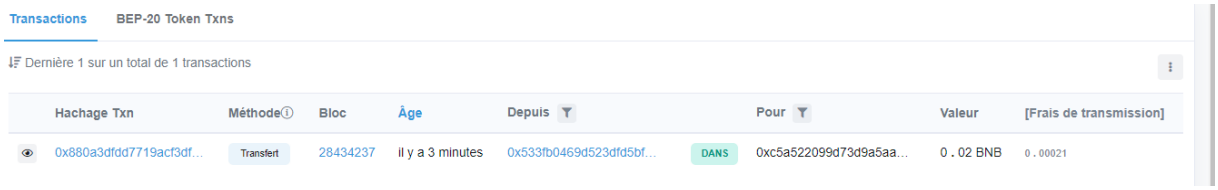
e) J’ai envoyé un mail pour recevoir mes premiers ETH.

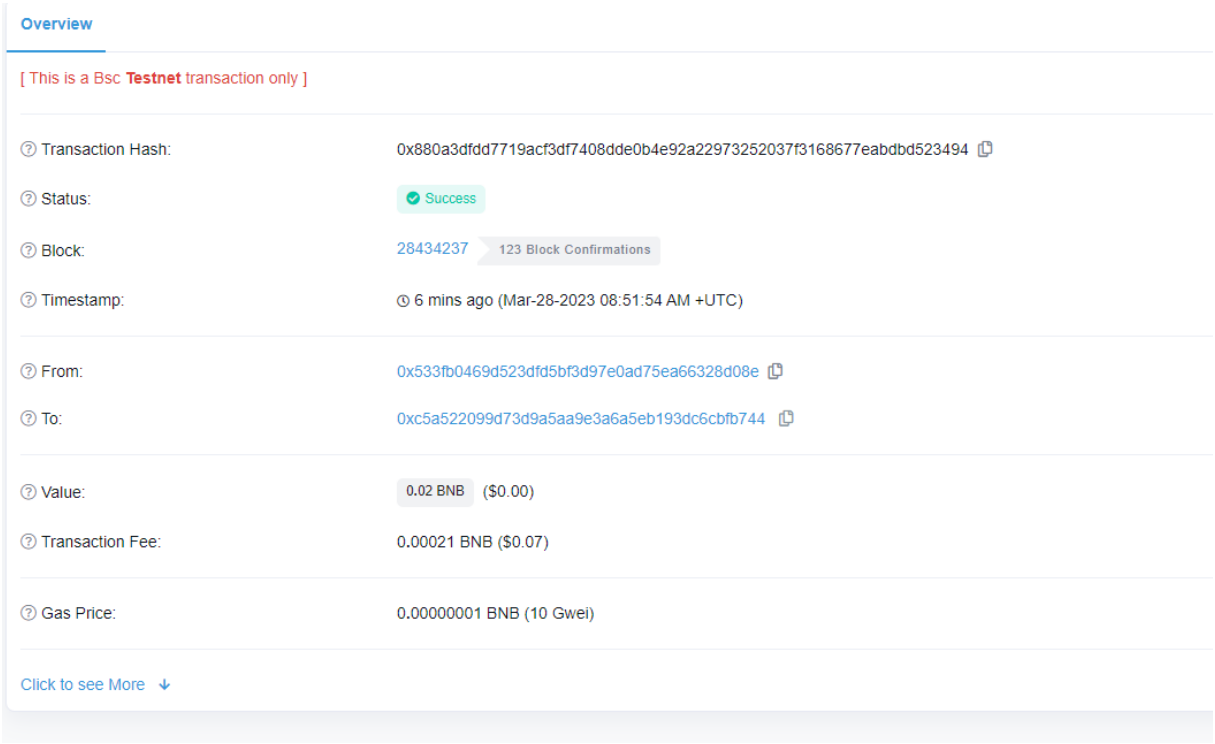


f) J’ai bien reçu mes premiers ETH :

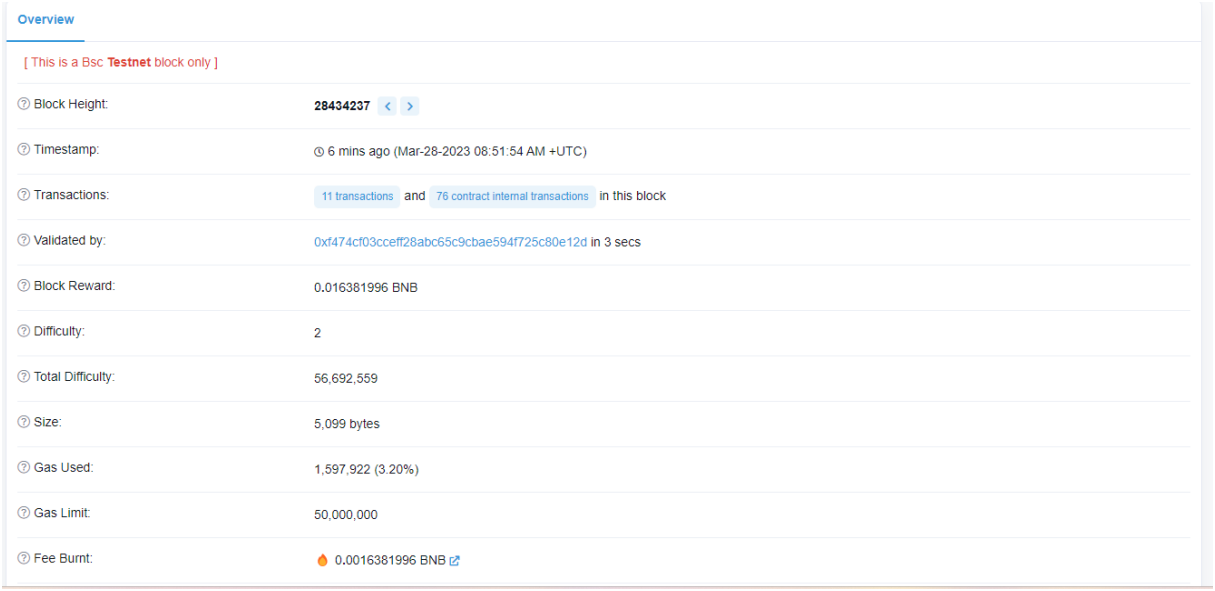


Voici les détails de la transaction ci-dessous :

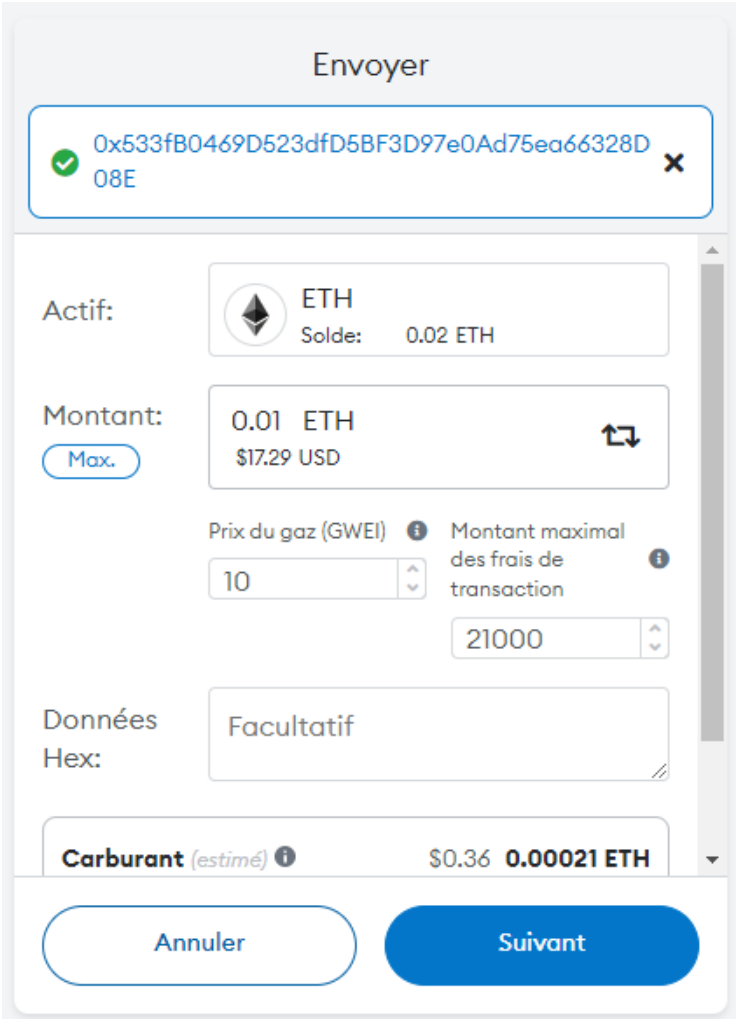


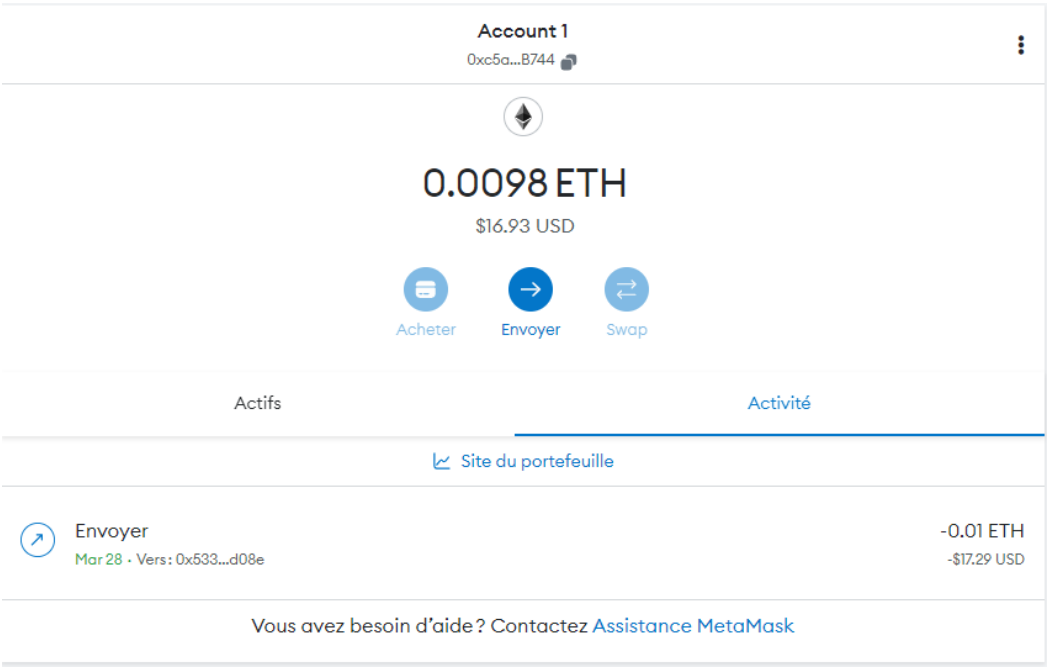


g) Ci-dessous nous avons le numéro de Block de la transaction. Ci-dessous les détails :

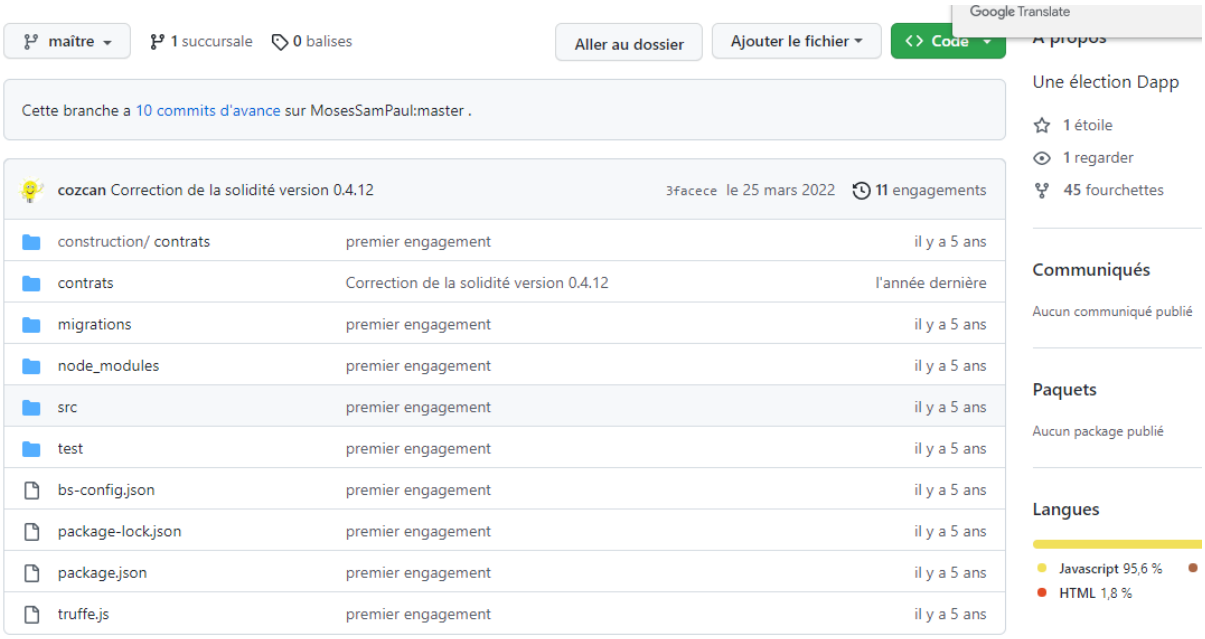


h) J’ai générer ma première transaction en envoyant 0.01 ETH :





i)j) IDE Remix a bien été ouvert et le code source de notre premier smart contract a été récupérer :



maître · 1 succursale · 0 balises · Aller au dossier · Ajouter le fichier · Code

Cette branche a 10 commits d'avance sur MosesSamPaul:master .

cozcan Correction de la solidité version 0.4.12 — 3facece le 25 mars 2022 — 11 engagements

construction/ contrats	premier engagement	il y a 5 ans
contrats	Correction de la solidité version 0.4.12	l'année dernière
migrations	premier engagement	il y a 5 ans
node_modules	premier engagement	il y a 5 ans
src	premier engagement	il y a 5 ans
test	premier engagement	il y a 5 ans
bs-config.json	premier engagement	il y a 5 ans
package-lock.json	premier engagement	il y a 5 ans
package.json	premier engagement	il y a 5 ans
truffe.js	premier engagement	il y a 5 ans

Une élection Dapp

1 étoile · 1 regarder · 45 fourchettes

Communiqués

Aucun communiqué publié

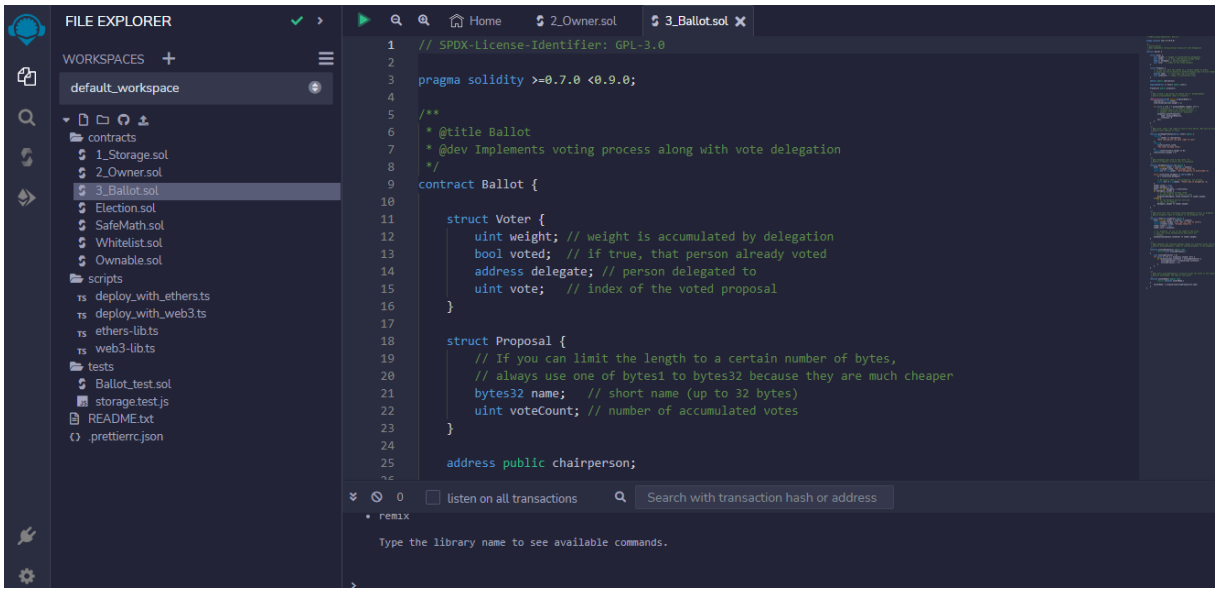
Paquets

Aucun package publié

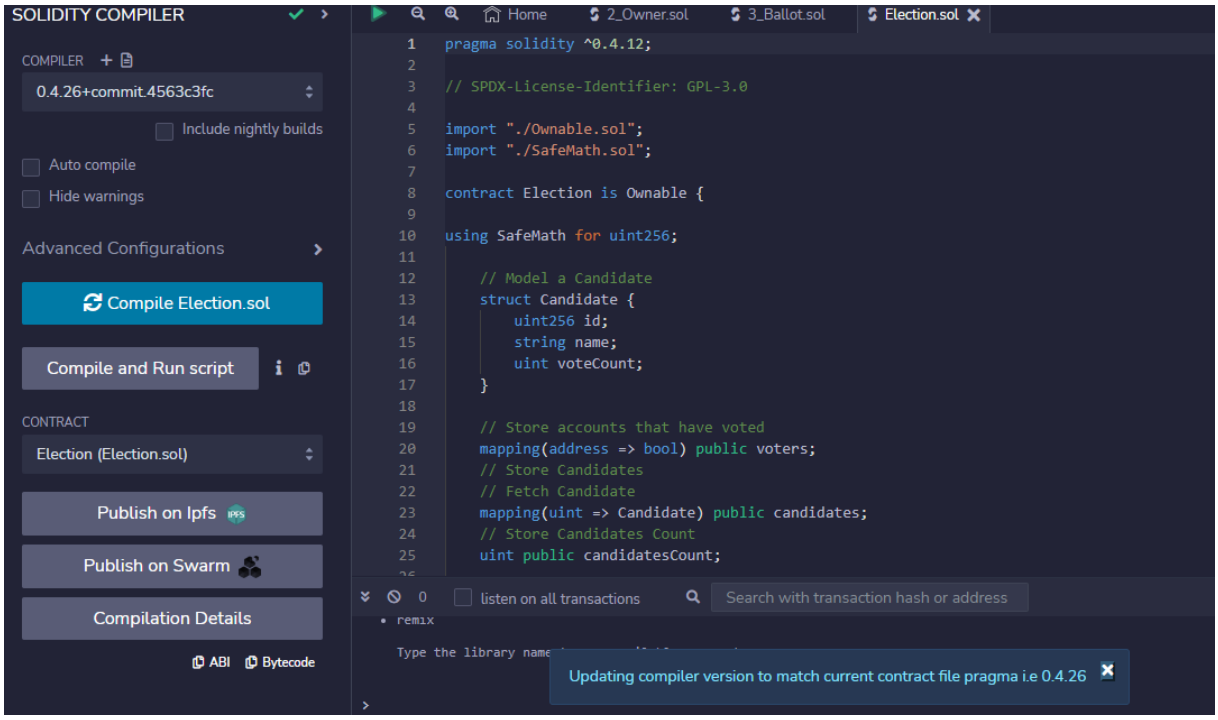
Langues

Javascript 95,6 % · HTML 1,8 %

k) L'ensemble des fichiers Solidity ont été ajouté sur notre environnement Remix :



I) Le smart contract « Election » a bien été compilé :



ABI :


```
[
	{
		"constant": false,
		"inputs": [
			{
				"name": "_candidateId",
				"type": "uint256"
			}
		],
		"name": "vote",
		"outputs": [],
		"payable": false,
		"stateMutability": "nonpayable",
		"type": "function"
	},
	{
		"constant": true,
		"inputs": [],
		"name": "candidatesCount",
		"outputs": [
			{
				"name": "",
				"type": "uint256"
			}
		],
		"payable": false,
```

```
        "stateMutability": "view",
        "type": "function"
},
{
        "constant": true,
        "inputs": [
                {
                        "name": "",
                        "type": "uint256"
                }
        ],
        "name": "candidates",
        "outputs": [
                {
                        "name": "id",
                        "type": "uint256"
                },
                {
                        "name": "name",
                        "type": "string"
                },
                {
                        "name": "voteCount",
                        "type": "uint256"
                }
        ],
        "payable": false,
        "stateMutability": "view",
        "type": "function"
},
{
        "constant": false,
        "inputs": [
                {
                        "name": "_name",
                        "type": "string"
                }
        ],
        "name": "addCandidate",
        "outputs": [],
        "payable": false,
        "stateMutability": "nonpayable",
        "type": "function"
},
{
        "constant": true,
        "inputs": [],
        "name": "owner",
        "outputs": [
                {
```

```
                        "name": "",
                        "type": "address"
                }
        ],
        "payable": false,
        "stateMutability": "view",
        "type": "function"
},
{
        "constant": true,
        "inputs": [
                {
                        "name": "",
                        "type": "address"
                }
        ],
        "name": "voters",
        "outputs": [
                {
                        "name": "",
                        "type": "bool"
                }
        ],
        "payable": false,
        "stateMutability": "view",
        "type": "function"
},
{
        "constant": false,
        "inputs": [
                {
                        "name": "newOwner",
                        "type": "address"
                }
        ],
        "name": "transferOwnership",
        "outputs": [],
        "payable": false,
        "stateMutability": "nonpayable",
        "type": "function"
},
{
        "anonymous": false,
        "inputs": [
                {
                        "indexed": true,
                        "name": "_candidateId",
                        "type": "uint256"
                }
        ],
```

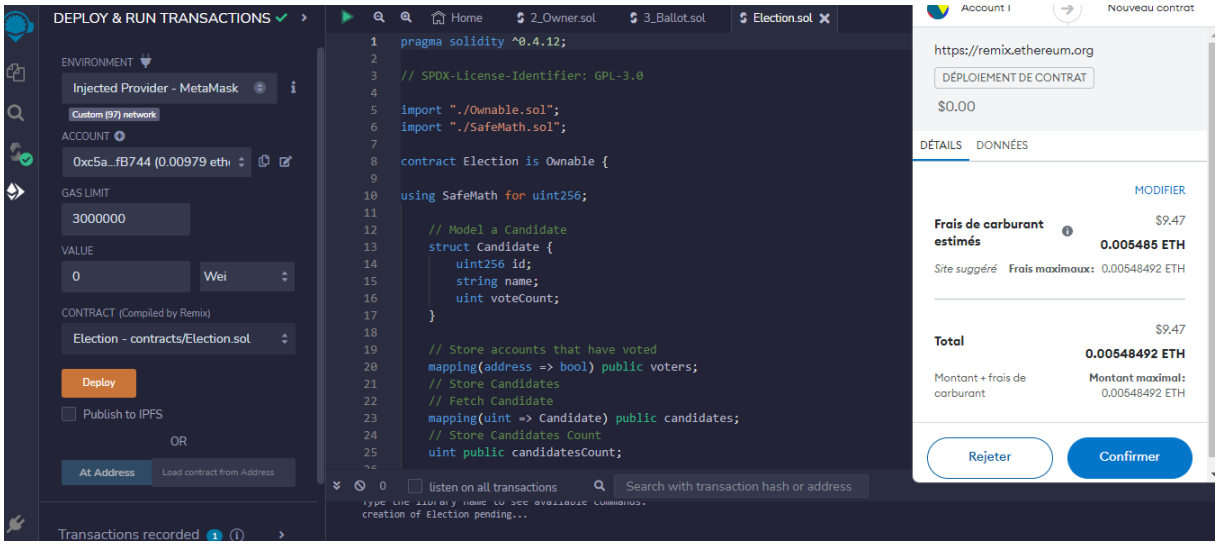
```
            "name": "votedEvent",
            "type": "event"
        },
        {
            "anonymous": false,
            "inputs": [
                {
                    "indexed": true,
                    "name": "previousOwner",
                    "type": "address"
                },
                {
                    "indexed": true,
                    "name": "newOwner",
                    "type": "address"
                }
            ],
            "name": "OwnershipTransferred",
            "type": "event"
        }
]
```

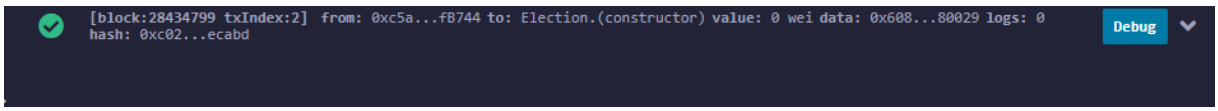

Bytecode :


```
608060405233600080610100a81548173ffffffffffffffffffffffffffffffffffffffff021916908373ffffffffffffffffffffffffffffffffffffffff160217905550610897806100536000396000f3006080604052600436106100835760003557c0100000000000000000000000000000000000000000000000000000000900463ffffffff1680630121b93f146100885780632d35a8a2146100b55780633477ee2e146100e0578063462e91ec146101945780638da5cb5b146101fd578063a3ec138d14610254578063f2fde38b146102af575b600080fd5b34801561009457600080fd5b506100b3600480360381019080803590602001909291905050506102f2565b005b3480156100c157600080fd5b506100ca610415565b6040518082815260200191505060405180910390f35b3480156100ec57600080fd5b506100f561041b565b6040518084815260200180602001838152602001828103825284818151815260200191508051906020019080838360005b8381101561015757808201518184015260208101905061013c565b50505050905090810190601f1680156101845780820380516001836020036101000a031916815260200191505b509450505050505060405180910390f35b3480156101a057600080fd5b506101fb600480360381019080803590602001908201803590602001908080601f0160208091040260200160405190810160405280939291908181526020018383808284378201915050505050509192919290505050506104dd565b005b34801561020957600080fd5b5061021261055a565b604051808273ffffffffffffffffffffffffffffffffffffffff1673ffffffffffffffffffffffffffffffffffffffff16815260200191505060405180910390f35b34801561026057600080fd5b506102956004803603810190808035906020019092919050505061057f565b604051808215151515815260200191505060405180910390f35b3480156102bb57600080fd5b506102f0600480360381019080803573ffffffffffffffffffffffffffffffffffffffff169060200190929190505050610599565b005b600160003373ffffffffffffffffffffffffffffffffffffffff1673ffffffffffffffffffffffffffffffffffffffff168152602001908152602001600020600090549061010a900460ff1615151561034b57600080fd5b600081118015610355d57506003548111155b151561036857600080fd5b600180600033373ffffffffffffffffffffffffffffffffffffffff1673ffffffffffffffffffffffffffffffffffffffff168152602001908152602001600020600020160008154809291906001019190505550606060405190810160405280603548152602001828054600181600116156101000203166002900480601f016020809104026020016040519081016040528092919081815260200182805460018160011615610100020316600290048015610cd5780601f106104a257610100808354040283529160200191610cd565b820191906000526020600020905b8154815290600101906020018083116104b057829003601f168201915b505050505090809060200154905083565b600360008154809291906001019190505550606060405190810160405280603548152602001828152602001600081525060026000600354815260200190815260200160002060008201518160000155602082015181600101908051906020019061054992919061076c6565b506040820151816002015590505050565b60008090549061010a900473ffffffffffffffffffffffffffffffffffffffff1681565b600160205280600052604060002060009150549061010a90046
```

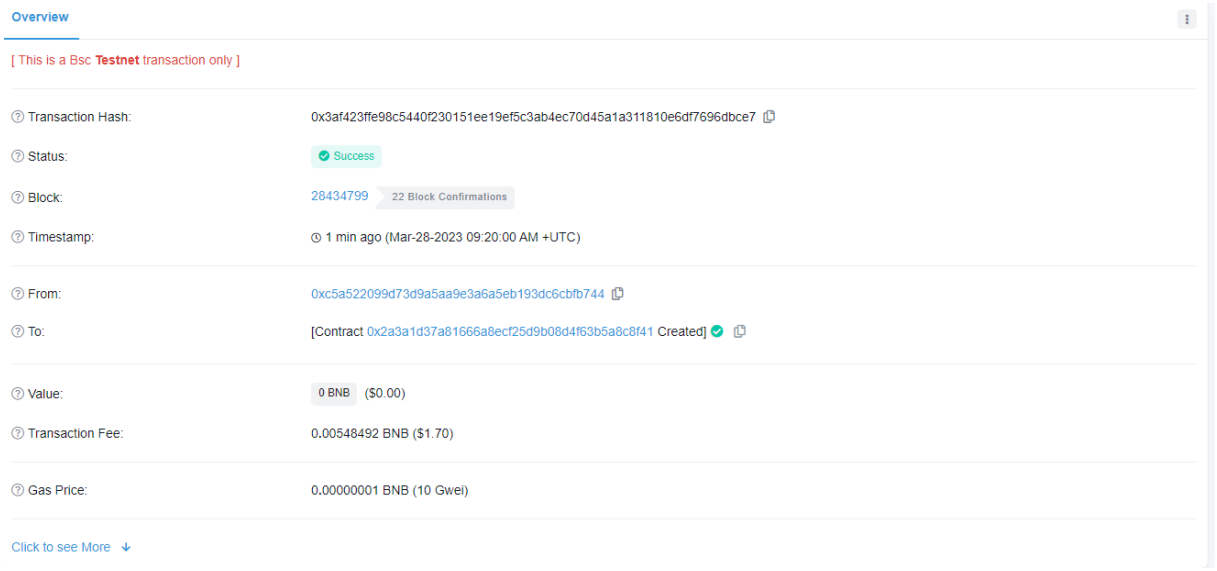
0ff1681565b60008090549061010000a900473ffffffffffffffffffffffffffffffffffffffff1673ffffffffffffffffffffffffffffffffffffffff1633
73ffffffffffffffffffffffffffffffffffffffff161415156106635760405117f08c379a0000000000000000000000000000000000000000
0000000000000000000000815260040180806020018281038252601881526020018071f4e6f74206175746f72697a6564
206f7065726174696f6e0000000000000000815250602001915050604051809103901fd5b600073ffffffffffffffffffffffff
ffffffffff168173ffffffffffffffffffffffffffffffffffffffff161415156107085760405117f08c379a0000000000000000000000000
0000000000000000000000000000000000000815260040180806020018281038252601981526020018071f41646472657373
2073686f756c646e2774206265207a65726f0000000000000000815250602001915050604051809103901fd5b8073ffffffffff
ffffffffffffffffffffffffff166000809054906101000a900473ffffffffffffffffffffffffffffffffffffffff1673ffffffffffffffffffffffffff
ffffffff167f8be0079c531659141344cd1fd0a4f28419497f9722a3daafe3b4186f6b6457e060405160405180910390a380
6000806101000a81548173ffffffffffffffffffffffffffffffffffffffff02191690837 3ffffffffffffffffffffffffffffffff16021790555
050565b828054600181600116156101000203166002900490600052602060002090601f016020900481019282601f106
1080757805160ff191683800117855561083556 5b8280016001018555821561083557918201 5b828111156108345782 5
1825591602001919060010190610819565b5b509050610842919061084656 5b5090565b6108689190 5b808211156108
64576000816000905550600101610 84c565b5090565b905600a165627a7a72305820c2b1daaf1ae81292340b32ac261cc
a45ecfd2e95a4ad3032438fcba078984ae80029

m) Déploiement du smart contract « Election.sol »

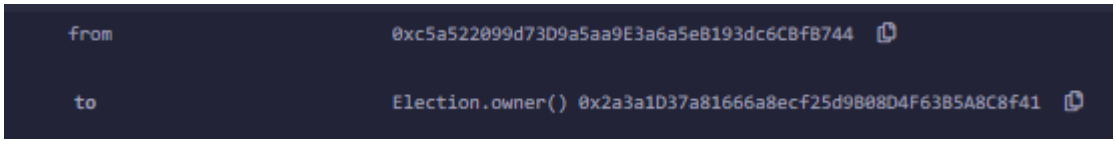




Ci-dessous les détails de la transaction :

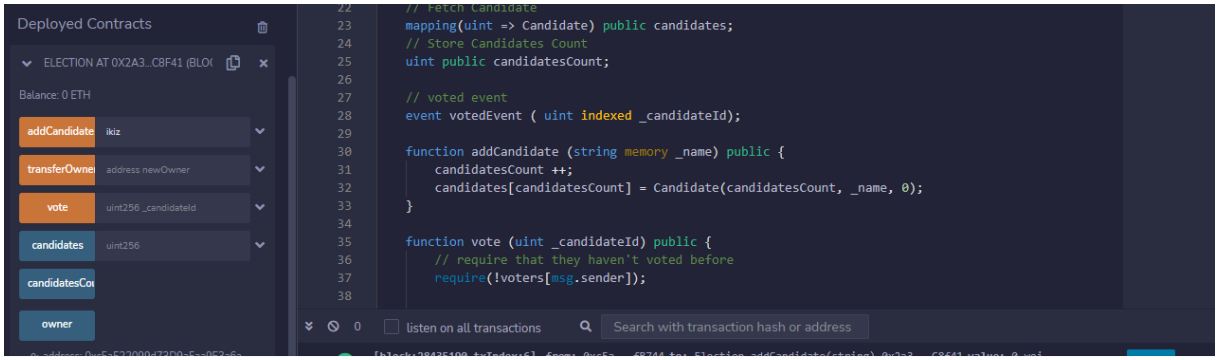


On a des frais de transaction plus élevé que les transactions précédentes. Les frais de transaction (Transaction fees) sont des frais payés pour effectuer une transaction sur un réseau de blockchain. Ces frais sont payés pour rémunérer les mineurs qui valident et permettent la transaction sur le réseau. Les frais sont plus élevés lors d'un déploiement car les mineurs ont plus de mal pour permettre la transaction.

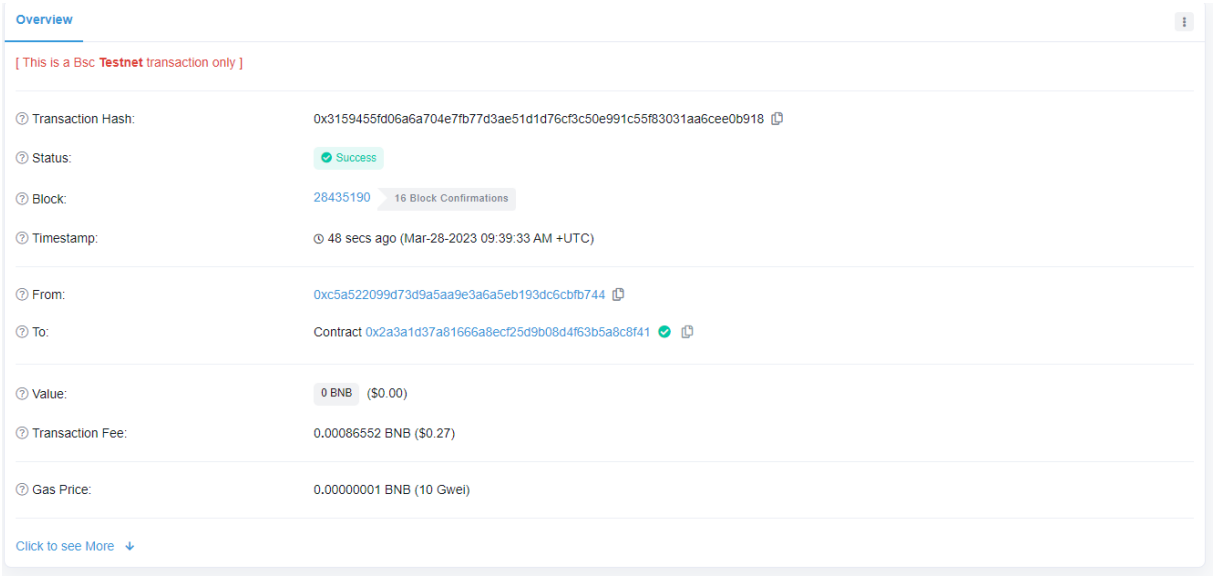


0x2a3a1D37a81666a8ecf25d9B08D4F63B5A8C8f41

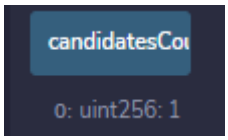
n) Nom du premier candidat ajouté (mon nom de famille) :

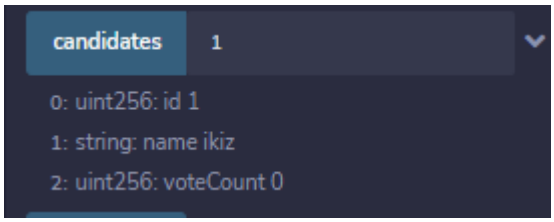


o) Voici ci-dessous les détails de la transaction :

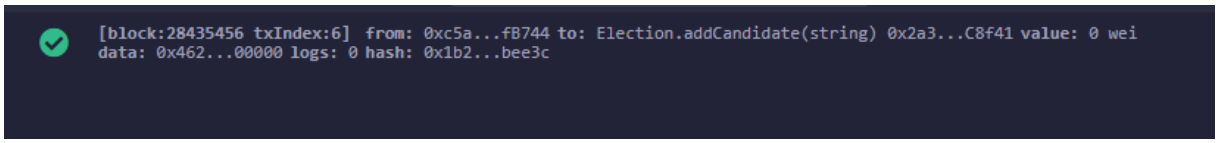


p) Voici ci-dessous la valeur de mon CandidateID ainsi que le détail :

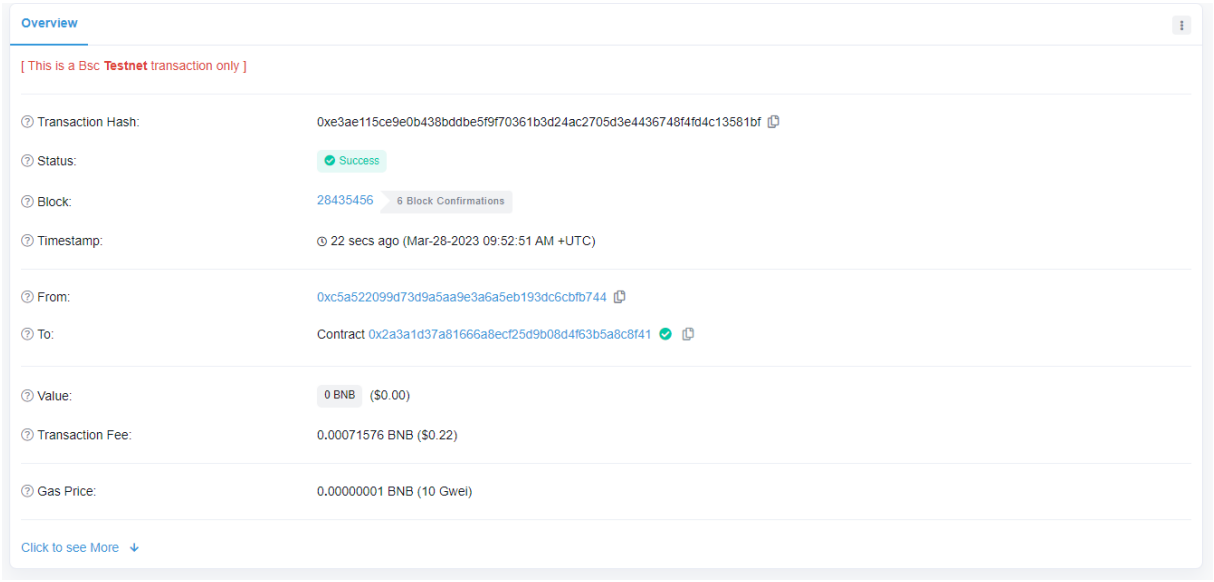




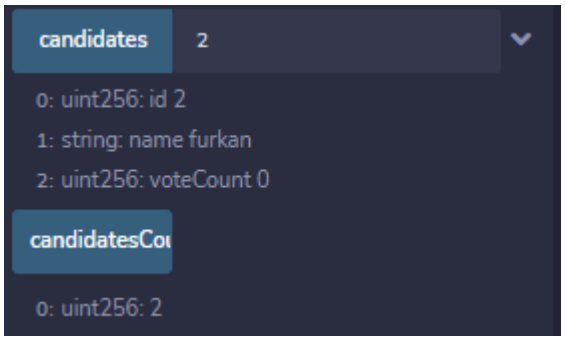
q) L'ajout d'un second candidat a été effectué :



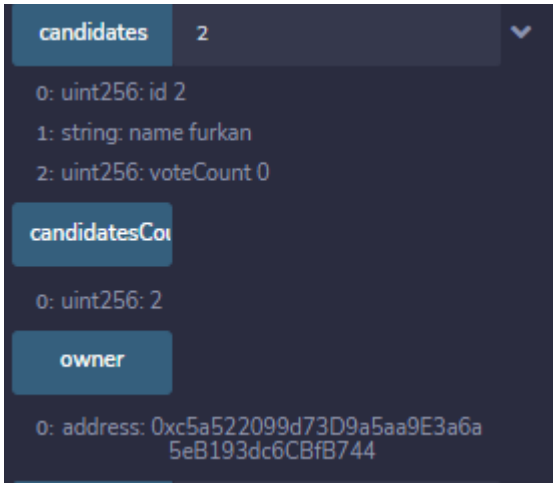
Voici les détails de la transaction :



r) Voici ci-dessous la valeur du second CandidateID ainsi que le détail :



s) En dessous d'owner l'adresse du propriétaire du contrat qui est le mien :



t) Réalisation du premier vote voici ci-dessous les détails de la transaction :

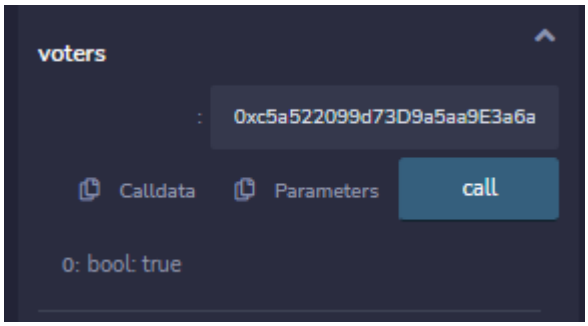
Overview Logs (1)

[This is a Bsc **Testnet** transaction only]

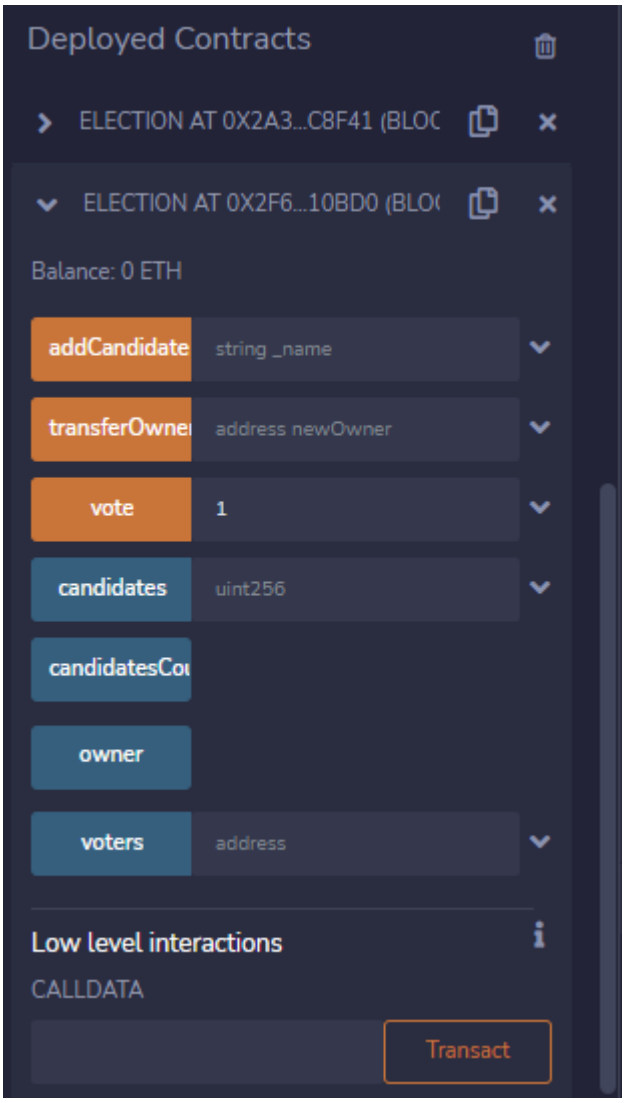
Transaction Hash:	0x8abbd2399195d956da8553db60edac96467fd0f75248baed5f52b92811b74867
Status:	Success
Block:	28435572 4 Block Confirmations
Timestamp:	13 secs ago (Mar-28-2023 09:58:39 AM +UTC)
From:	0xc5a522099d73d9a5aa9e3a6a5eb193dc6cbfb744
To:	Contract 0x2a3a1d37a81666a8ecf25d9b08d4f63b5a8c8f41
Value:	0 BNB ($0.00)
Transaction Fee:	0.00066241 BNB ($0.20)
Gas Price:	0.00000001 BNB (10 Gwei)

Click to see More

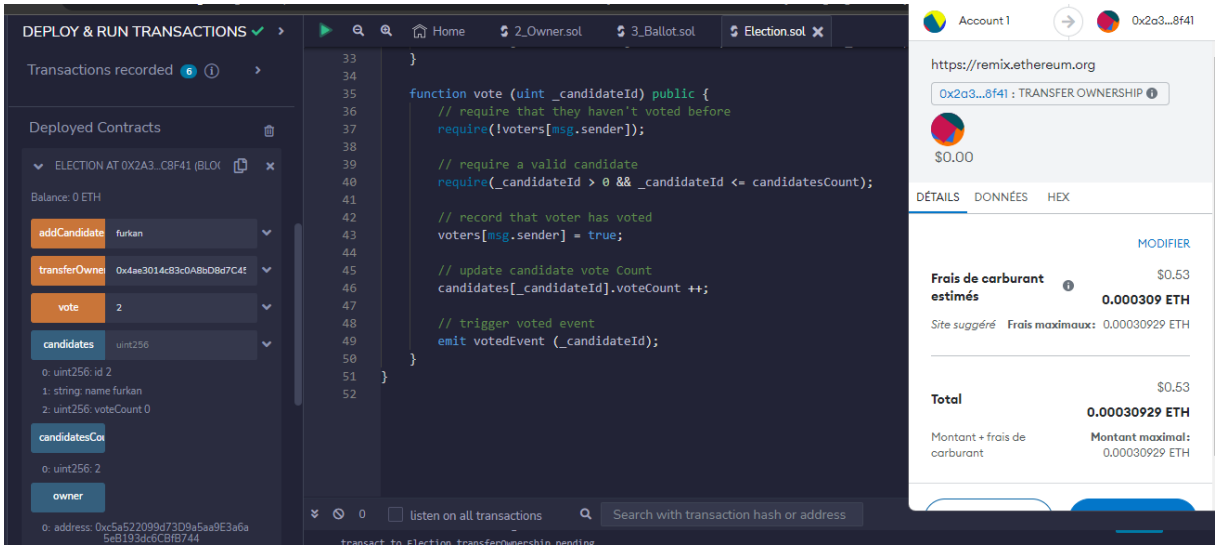
u) Le vote a bien été pris en compte :



v) J’ai interagis avec le contrat de mon camarade



w) J'ai bien effectuer le transfert de la propriété à mon camarade :



x) Pour sécuriser il faudrait mettre la fonction addCandidate en private.

y) Voici la modification du code :

```
function addCandidate (string memory _name) private {
    candidatesCount ++;
    candidates[candidatesCount] = Candidate(candidatesCount, _name, 0);
}
```