



SONIC PLATFORM

A platform that collects user-related data about the crypto sector in exchange for a reward in Sonic Tokens

Features implemented

- User authentication through Metamask, check profile info
- Test Smart Contract with Brownie and Ganache
- Deploy on Remix Sonic Smart Contract (ERC20)
- Fill out the Survey and receive a reward in Sonic Tokens
- Get the transaction receipt after the survey submission
- Save all the surveys in Mongo DB Database

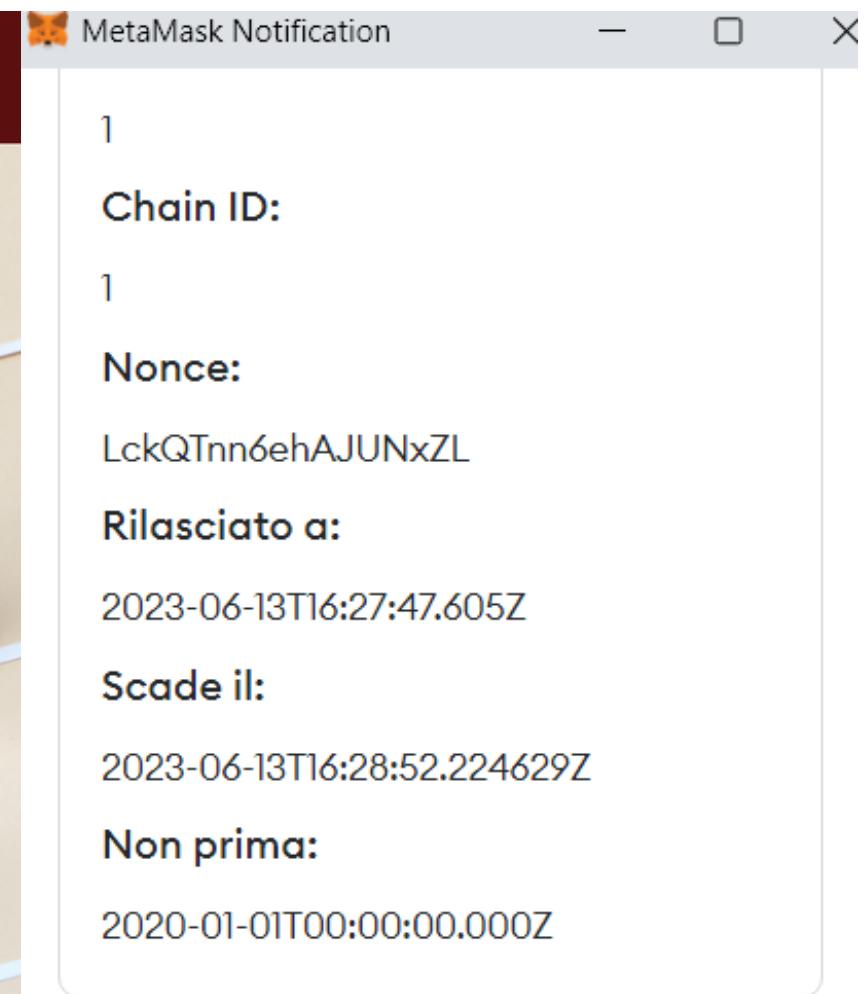


Login with Metamask



After clicking on the "login with Metamask" button, you will have the opportunity to confirm your Metamask account access to the platform.

(Remember to add Metamask extension on your browser)



⚠ Deceptive site request.
The site you're attempting to sign into doesn't match the domain in the request. Proceed with caution.

Annulla

Sign-In

Welcome message and profile info

SONIC

Welcome Web3 User!

Eth Address: 0xEE958EE253e5eE92DE2975a43a317aef32f34324

[Logout](#)
[My profile](#)



Welcome 0xEE958EE253e5eE92DE2975a43a317aef32f34324 - Survey

Eth Address: 0xEE958EE253e5eE92DE2975a43a317aef32f34324

Session auth info

defi.finance wants you to sign in with your Ethereum account:
0xEE958EE253e5eE92DE2975a43a317aef32f34324

Please confirm

message

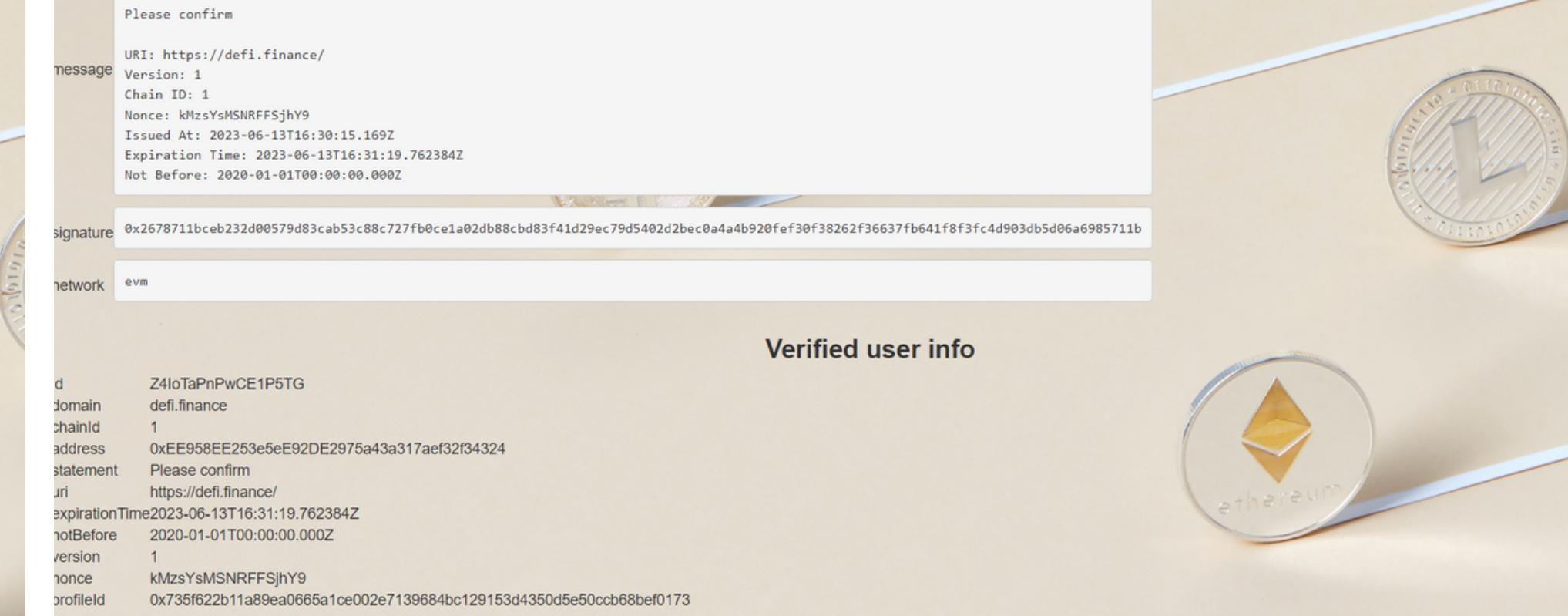
URI: <https://defi.finance/>
Version: 1
Chain ID: 1
Nonce: kMzsYsMSNRFFSjhY9
Issued At: 2023-06-13T16:30:15.169Z
Expiration Time: 2023-06-13T16:31:19.762384Z
Not Before: 2020-01-01T00:00:00.000Z

signature 0x2678711bceb232d00579d83cab53c88c727fb0ce1a02db88cbd83f41d29ec79d5402d2bec0a4a4b920fef30f38262f36637fb641f8f3fc4d903db5d06a6985711b

network evm

Verified user info

d Z4loTaPnPwCE1P5TG
domain defi.finance
chainId 1
address 0xEE958EE253e5eE92DE2975a43a317aef32f34324
statement Please confirm
uri <https://defi.finance/>
expirationTime 2023-06-13T16:31:19.762384Z
notBefore 2020-01-01T00:00:00.000Z
version 1
nonce kMzsYsMSNRFFSjhY9
profileId 0x735f622b11a89ea0665a1ce002e7139684bc129153d4350d5e50ccb68bef0173



After logging in, a welcome message will appear with your eth address and you will be able to see the account info at this endpoint: http://127.0.0.1:8000/my_profile/

Test Smart Contract with Brownie and Ganache



```
BrownieTestingProject is the active project.

Launching 'ganache-cli.cmd --port 8545 --gasLimit 12000000 --accounts 10 --hardfork istanbul --mnemonic brownie'...

Running 'scripts\test_deploy_Sonic.py::main'...
Transaction sent: 0x5f5d2733720755dad8a838e49c674f947f257c83a33ee042b3ac210a896229f8
  Gas price: 0.0 gwei  Gas limit: 12000000  Nonce: 0
Sonic.constructor confirmed  Block: 1  Gas used: 506569 (4.22%)
Sonic deployed at: 0x3194cBDC3dbcd3E11a07892e7bA5c3394048Cc87

=====
 test session starts =====
platform win32 -- Python 3.7.9, pytest-6.2.5, py-1.11.0, pluggy-1.0.0
rootdir: C:\Users\andre\OneDrive\Desktop\Progetto Ethereum\ethproject\brownie-testing
plugins: eth-brownie-1.19.3, hypothesis-6.27.3, forked-1.4.0, xdist-1.34.0, web3-5.31.3
collected 5 items

Launching 'ganache-cli.cmd --port 8545 --gasLimit 12000000 --accounts 10 --hardfork istanbul --mnemonic brownie'...

tests\unit\test_Sonic.py ......

===== 5 passed in 16.82s =====
Terminating local RPC client...
```

Thanks to Brownie and Ganache I tested Sonic Smart Contract inside VSCode with Python.

Deploy on Remix the Smart Contract Sonic (ERC20)



The screenshot shows the Remix IDE interface. On the left is the File Explorer sidebar with a tree view of the workspace. The root folder is 'default_workspace' containing 'contracts', 'artifacts', 'sonic.sol', 'scripts', 'tests', 'README.txt', and '.prettierrc.json'. The 'artifacts' folder is currently selected. The main central area is the code editor showing the Solidity smart contract 'sonic.sol'. The code defines an ERC20Interface and a Sonic contract that implements it. The Sonic contract has variables for symbol, name, decimals, and totalSupply, along with mappings for balances and allowances. It includes a constructor setting the symbol to 'SNC' and the name to 'Sonic'. At the bottom of the code editor, there are tabs for 'ethers.js' and 'remix', and a search bar. The bottom right of the interface has a command line input field.

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.17;

interface ERC20Interface {
    function totalSupply() external view returns (uint);
    function balanceOf(address account) external view returns (uint balance);
    function allowance(address owner, address spender) external view returns (uint remaining);
    function transfer(address recipient, uint amount) external returns (bool success);
    function approve(address spender, uint amount) external returns (bool success);
    function transferFrom(address sender, address recipient, uint amount) external returns (bool success);

    event Transfer(address indexed from, address indexed to, uint value);
    event Approval(address indexed owner, address indexed spender, uint value);
}

//Token contract

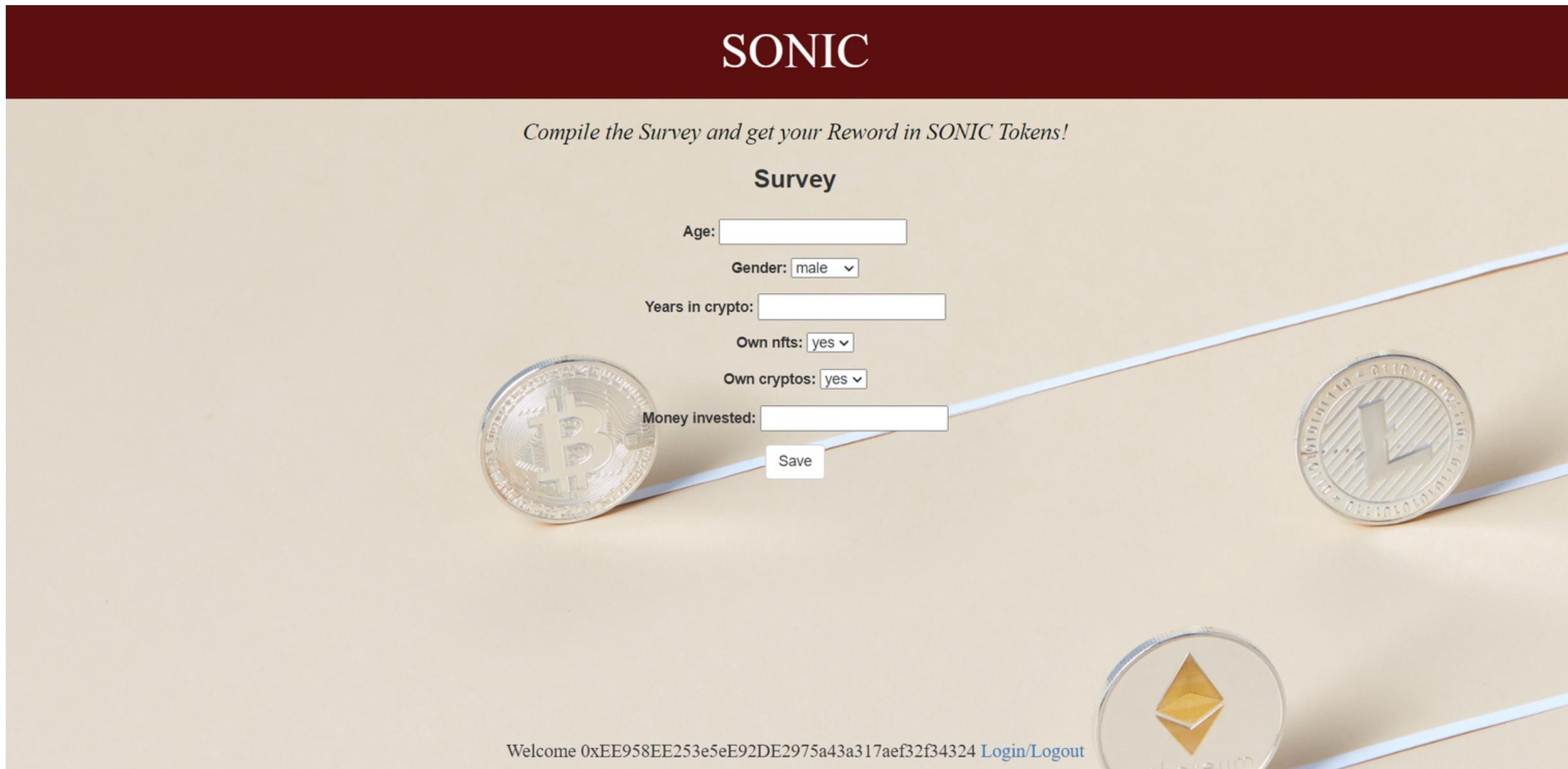
contract Sonic is ERC20Interface {
    string public symbol;
    string public name;
    uint8 public decimals;
    uint public _totalSupply;

    mapping(address => uint) balances;
    mapping(address => mapping(address => uint)) allowed;

    constructor() {
        symbol = "SNC";
        name = "Sonic";
    }
}
```

Thanks to Remix, a powerful toolset for developing, deploying, debugging, and testing Ethereum, I compiled and deploy Sonic Smart Contract.

Compile and submit the Survey



After filling out the survey, a transaction will send the Sonic Tokens that will arrive directly on the user's address.

The survey will be saved in the mongodb database and the transaction receipt will be print on the monitor

THANK YOU FOR YOUR ATTENTION

Link Git Hub and Pythonanywhere for the project:

https://github.com/andreagenos/ethereum_project.git

Admin User: Username - AndreaAdmin / Password - prova123



Andrea Genovese