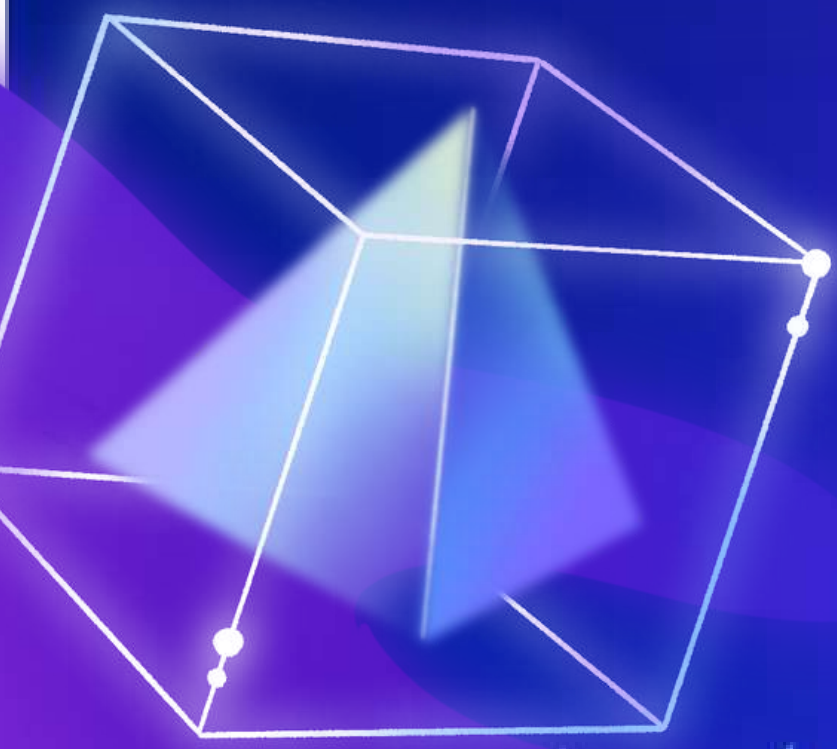
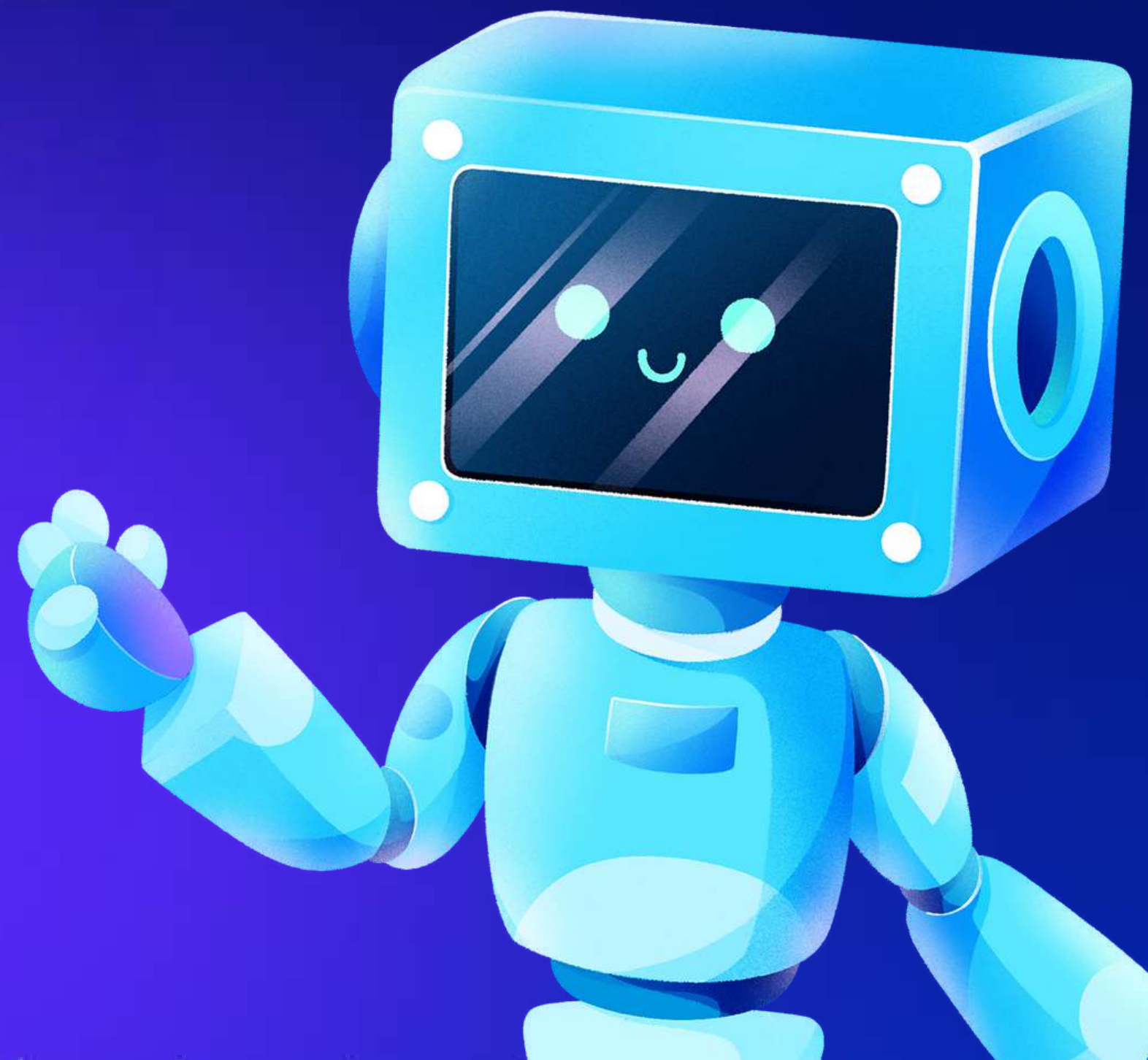




SMART CONTRACT HAPPYFOUND

Di Nieli Giorgia 13/02/2024



PRESENTAZIONE E SVILUPPO SMART CONTRACT “HAPPYFOUND”

Inizio la presentazione del mio progetto partendo dal nome che gli ho voluto assegnare, HappyFound, d'altronde quando parliamo di beneficenza stiamo facendo del bene e come in questo caso, una raccolta fondi a fin di bene credo che renda tutti fieri e felici anche solo per un istante.

Obiettivi

La nuova tecnologia su cui è basato lo smart contract “HappyFound” (Blockchain Ethereum) permetterà l'autogestione del contratto stesso, senza la necessità di un controllo o di terze parti e la massima sicurezza del sistema. Dato l'input dell'obiettivo da raggiungere verrà “lanciato” lo smart contract e ci basterà aspettare il suo raggiungimento e prelevare i fondi raccolti.

In poche parole, HappyFound terrà il conto dei donatori, del saldo totale, dello stato dell'obiettivo .

Ovviamente, esclusivamente il Manager sarà in grado di prelevare i fondi .



CODICE SORGENTE HAPPYFOUND

Vediamo nel dettaglio come funziona il nostro smart contract e i risultati che produrrà :

Ancor prima di distribuire il nostro contratto sarà necessario fornire un Input , ovvero l'obiettivo del fondo ... es. 100
Inoltre stabiliremo che l'indirizzo che ha distribuito lo smart contract è quello del manager e quindi solo quell'indirizzo potrà prelevare i fondi raccolti :

```
1  // SPDX-License-Identifier: MIT
2
3  pragma solidity ^0.8.0;
4
5  contract HappyFound {
6
7      //Definizione dati che utilizzeremo
8      address payable AddresManager;
9      uint Balance;
10     uint public Donors;
11     uint public Goal;
12     uint Donor;
13
14     // Calcoli necessari al funzionamento dello smart contract :
15
16     constructor(uint _EnterGoalFound) {    ⛊ infinite gas 177600 gas
17         //Definizione dati iniziali
18         AddresManager = payable(msg.sender);
19         Goal=_EnterGoalFound;
20     }
21 }
```



ACCOUNT

0x5B3...eddC4 (99.99999999%)

LIMITE DI GAS

3000000

VALORE

0

Wei

CONTRATTO

HappyFound - GIORGIA NIELI PROGE

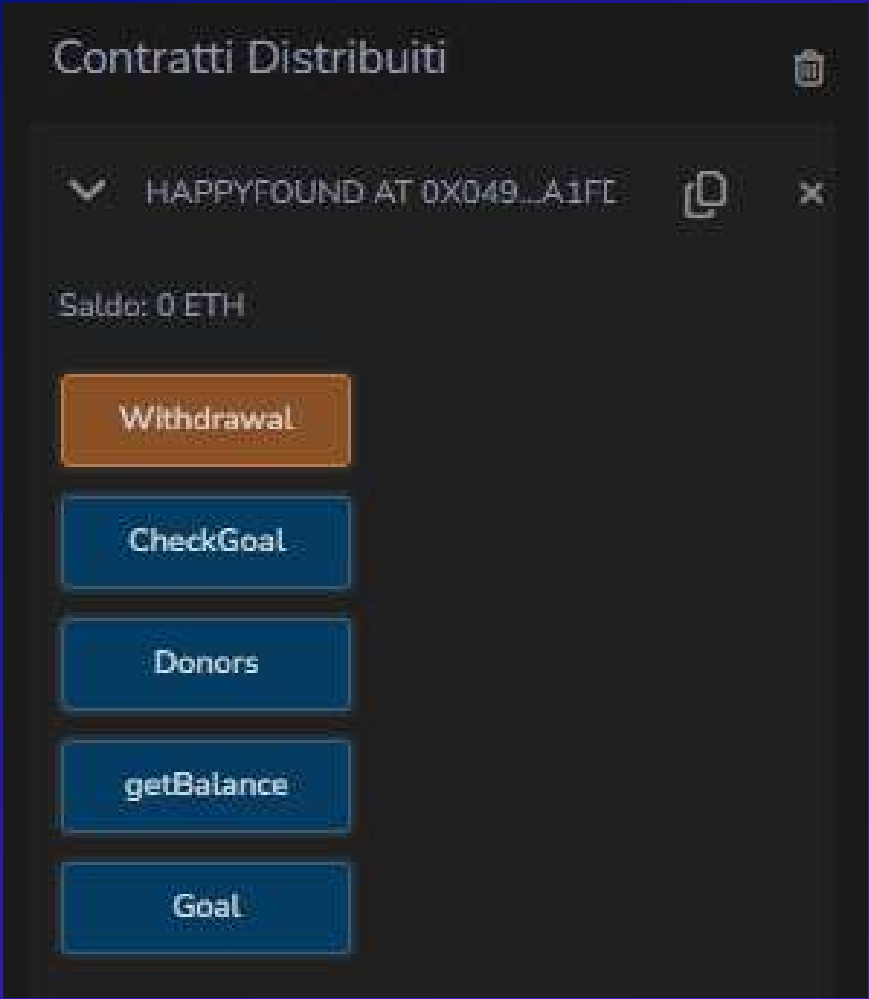
versione EVM: london

Distribuisci 100

☐ Pubblica su IPFS

All'Indirizzo 0x5B38Da6a701c568545dCfcE

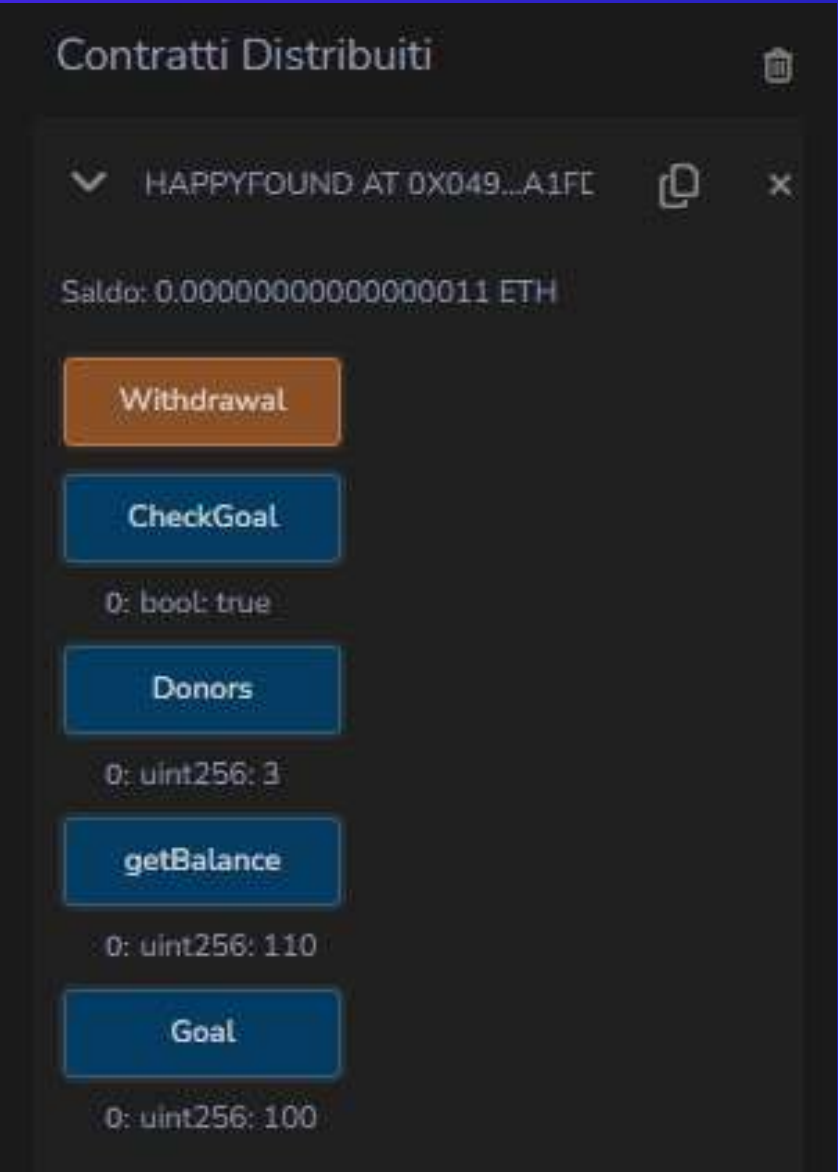
Entriamo nel vivo dell'azione e vediamo le nostre funzioni e variabili che producono dei risultati :



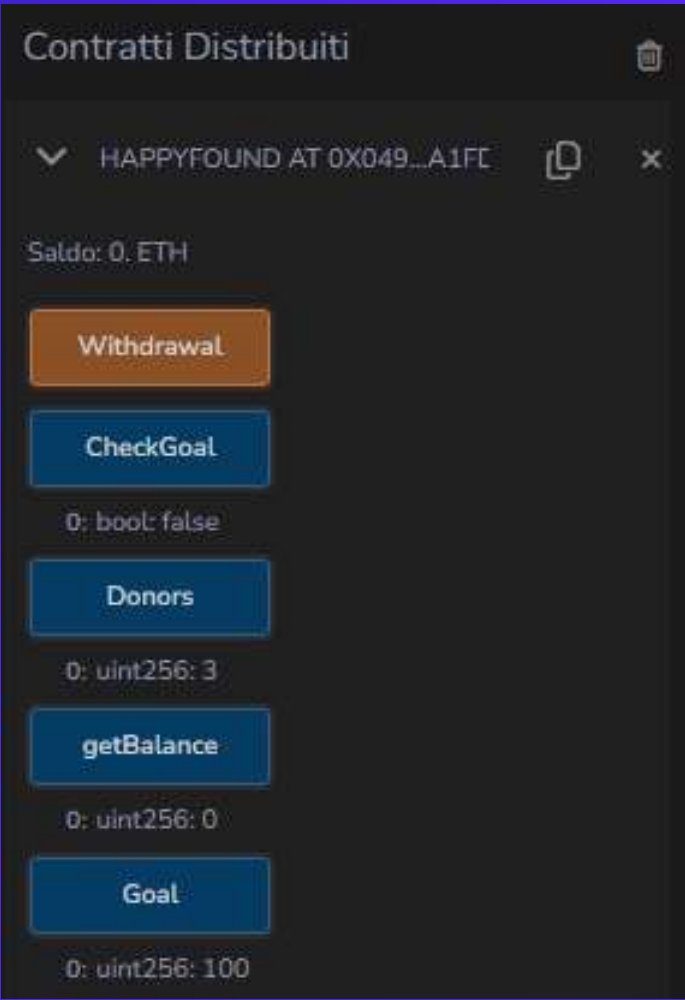
Sulla sinistra abbiamo la schermata iniziale del nostro contratto quindi :
Withdrawal = Prelievo del fondo , ovviamente realizzabile solo dall'indirizzo Manager prima stabilito
CheckGoal = Ci dirà "Vero" se l'obiettivo è stato raggiunto o "Falso" se l'obiettivo non è stato raggiunto
Donors = Numero totale dei donatori
getBalance = Saldo Fondo
Goal = Obiettivo fondo prima stabilito



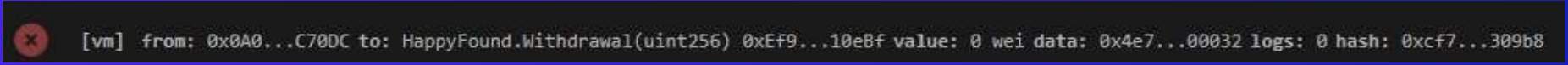
Facciamo un esempio pratico e supponiamo che gli utenti A, B , C donano un totale di 110 : Sulla schermata di Destra vediamo l'aggiornamento corretto delle voci:



Per il prelevamento invece basterà cliccare Withdrawal e se la richiesta proviene dall'indirizzo manager corretto , il contratto accrediterà i fondi e aggiornerà il saldo a 0 :



Se l'indirizzo non è del Manager , l'operazione sarà rilevata come un errore :




```
// SPDX-License-Identifier: MIT

pragma solidity ^0.8.0;

contract HappyFound {

    //Definizione dati che utilizzeremo
    address payable AddresManager;
    uint Balance;
    uint public Donors;
    uint public Goal;
    uint Donor;

    // Calcoli necessari al funzionamento dello smart contract :

    constructor(uint _EnterGoalFound) {    ⚡ Infinite gas 177660 gas
        //Definizione dati iniziali
        AddresManager = payable(msg.sender);
        Goal=_EnterGoalFound;
    }

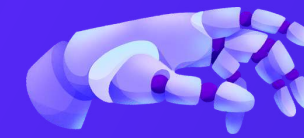
    receive() external payable {    ⚡ undefined gas
        // Fare Donazioni e Contare Totale Donatori
        require(msg.value>0);
        Donor = msg.value+Donor;
        Donors++;
    }

    function getBalance() public view returns (uint) {    ⚡ 317 gas
        // Visualizzare saldo del Fondo
        return address(this).balance;
    }

    function CheckGoal() public view returns (bool) {    ⚡ 2558 gas
        // Verificare raggiungimento obiettivo
        if (getBalance()>=Goal) {
            return true;
        }
        return false;
    }

    function Withdrawal() public {    ⚡ Infinite gas
        // Funzione di prelievo esclusivo per il Manager Found
        require(msg.sender==AddresManager);
        payable(msg.sender).transfer(address(this).balance);
    }

}
```



CURIOSITA' :

CODICE

SORGENTE

HAPPYFOUND

**Faremo tutte le prove e le verifiche necessarie insieme per migliorare e rendere completo il nostro
"contratto intelligente"
prima che venga Deployato definitivamente e quindi non più modificato.**

SMART CONTRACT HAPPYFOUND

Di Nieli Giorgia 13/02/2024

