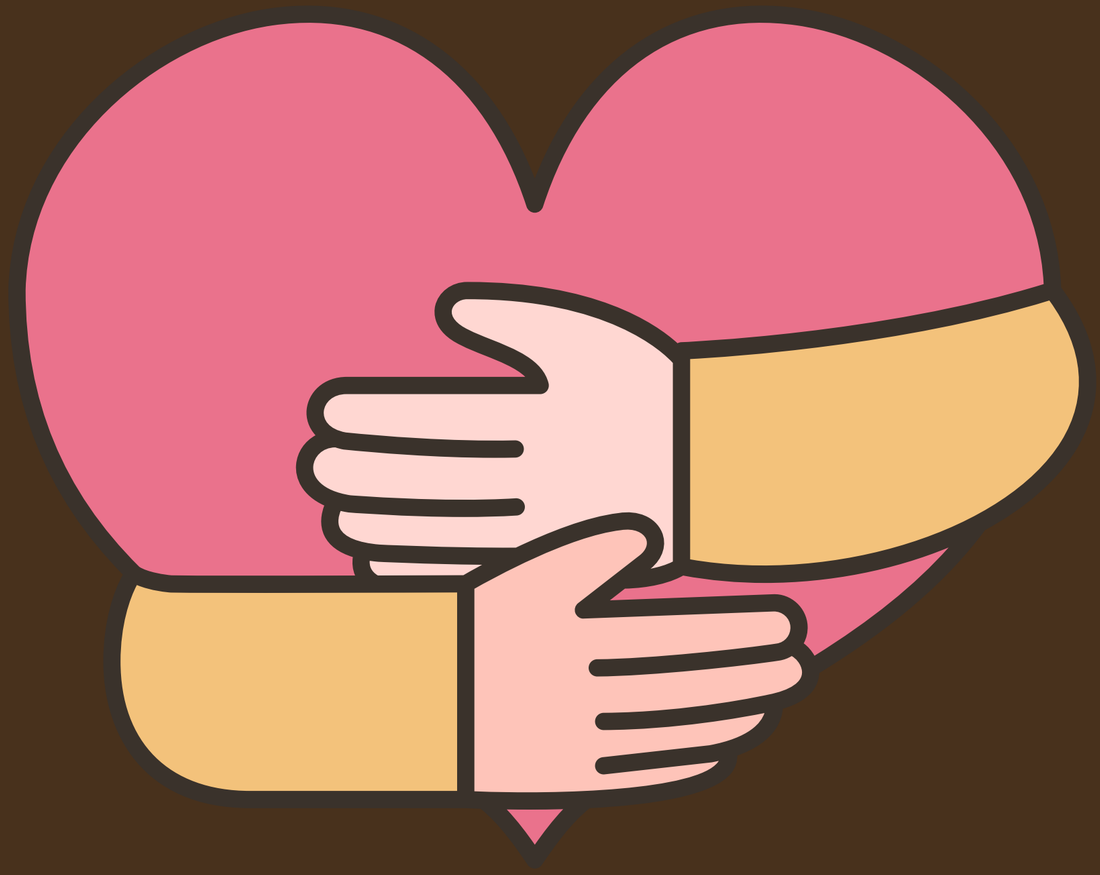
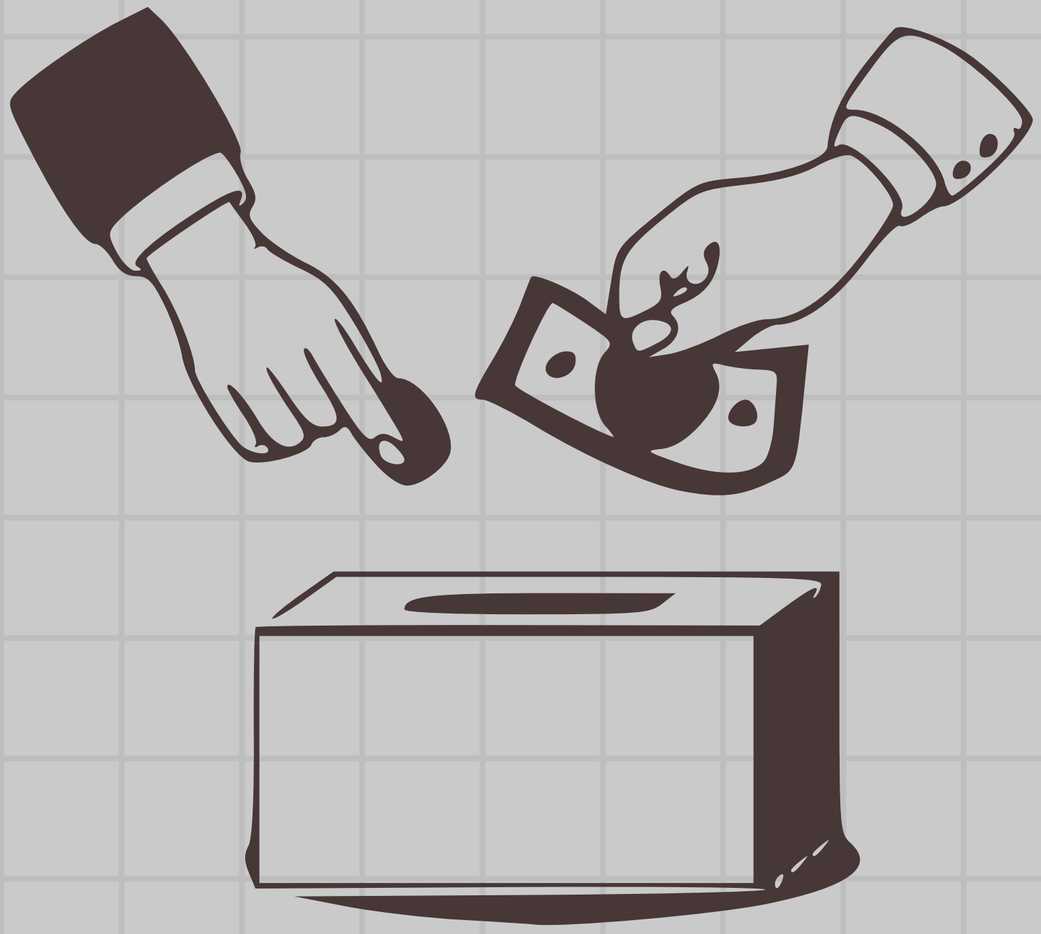
Charity dApp

Decentralized Charity Crowdfunding DApp



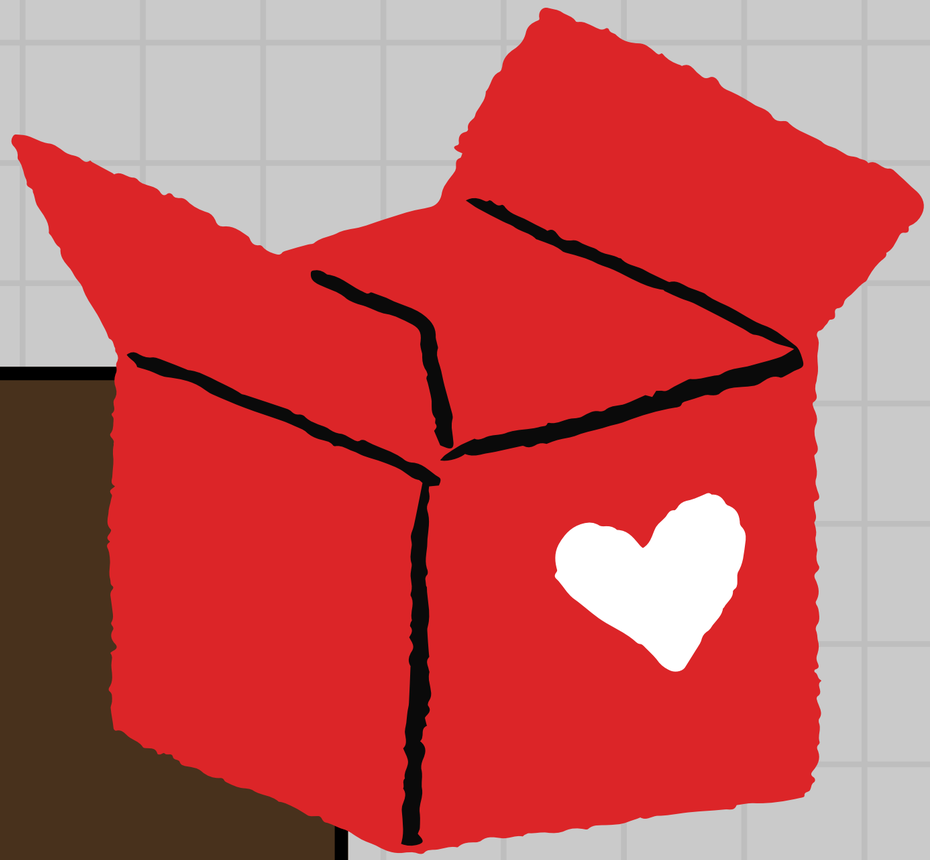
Oryngul Maratova, Danelya Bekzhan, Aigerim Kazbek
SE-2426



About the project

This project a decentralized crowdfunding application built on the Ethereum Sepolia test network.

Analyzing the problem



Today many crowdfunding platforms are centralized.
This means:

- users must trust a single company,
- transaction history is not always transparent,
- fees can be high,
- data can be changed or hidden.

Examples: GoFundMe, Patreon

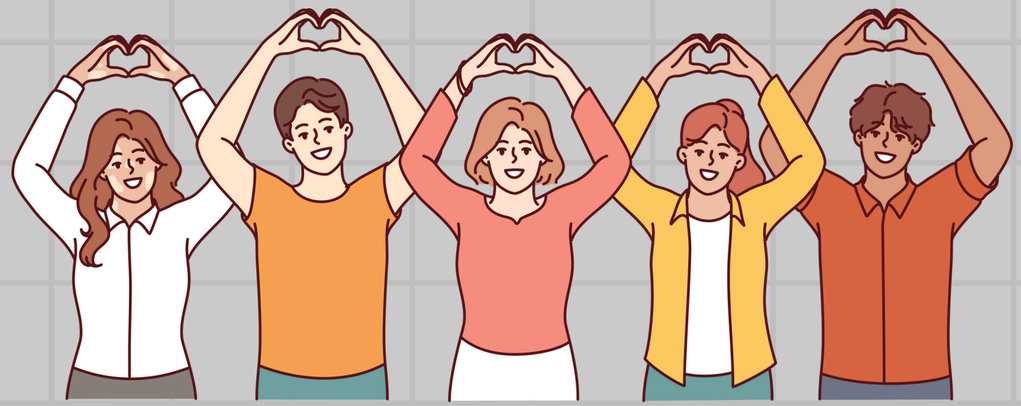
Idea of the Project

The idea is to create a decentralized charity platform where:

- all transactions are stored on blockchain,
- no central authority controls the funds,
- everyone can verify donations,
- users interact directly with smart contracts
- MetaMask wallet is integrated for secure login and transaction approval.

How It Works (Concept)

- A user creates a campaign with a goal and deadline.
- Users donate test ETH.
- Donors receive reward tokens automatically.
- After the deadline:
- if goal is reached - funds go to beneficiary,
- if goal is not reached - donors can refund.



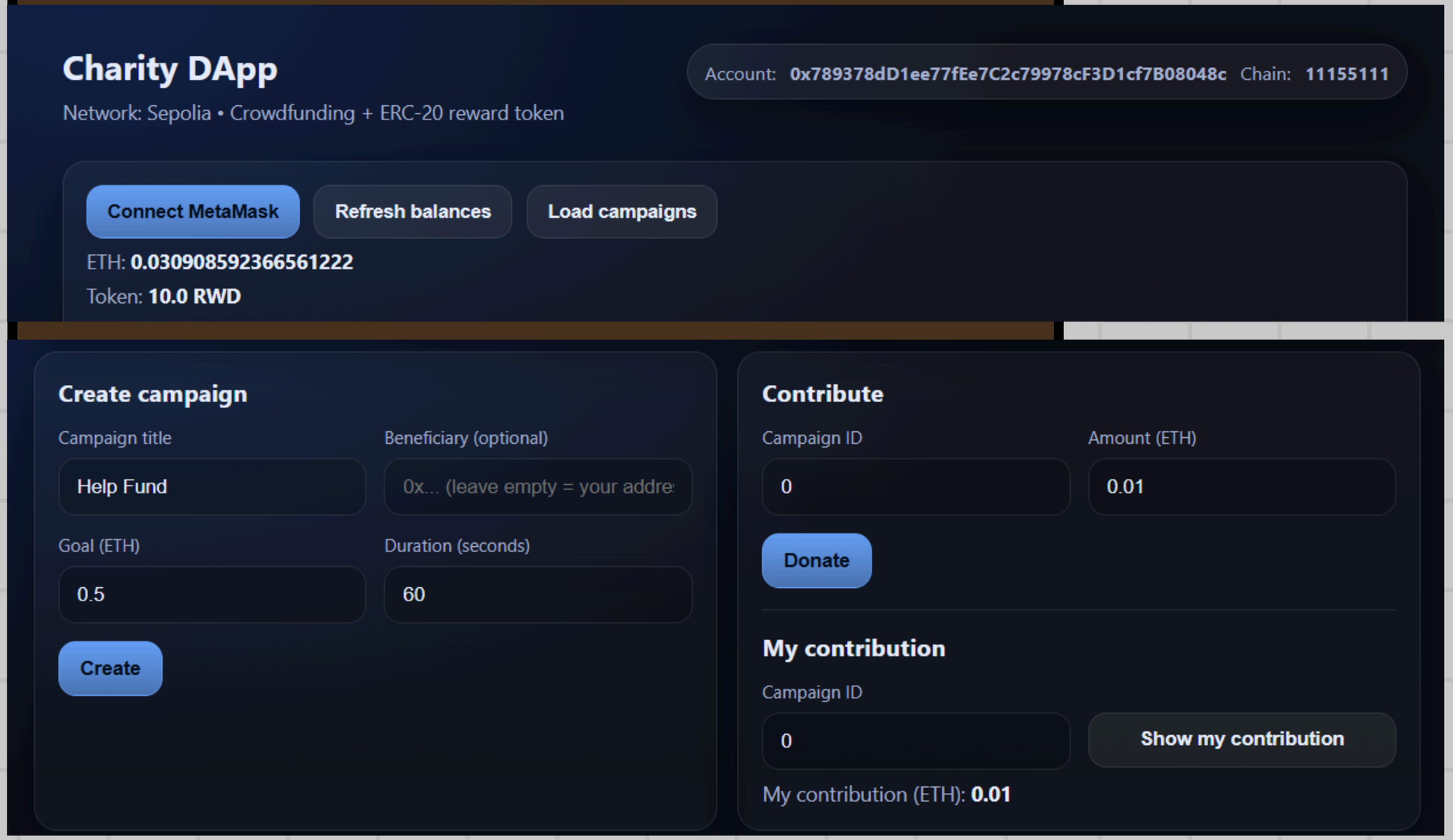
Key Features

Campaign creation

Automatic reward tokens

MetaMask integration

Results


Charity DApp
Network: Sepolia • Crowdfunding + ERC-20 reward token
Account: 0x789378dD1ee77fEe7C2c79978cF3D1cf7B08048c Chain: 11155111
Connect MetaMask
Refresh balances
Load campaigns
ETH: 0.030908592366561222
Token: 10.0 RWD
Create campaign
Campaign title
Help Fund
Beneficiary (optional)
0x... (leave empty = your addre
Goal (ETH)
0.5
Duration (seconds)
60
Create
Contribute
Campaign ID
0
Amount (ETH)
0.01
Donate
My contribution
Campaign ID
0
Show my contribution
My contribution (ETH): 0.01

Campaigns

#0 Help Fund
goal: 0.5 ETH | raised: 0.01 ETH
status: active=false finalized=true successful=false

Finalize / Refund

Campaign ID

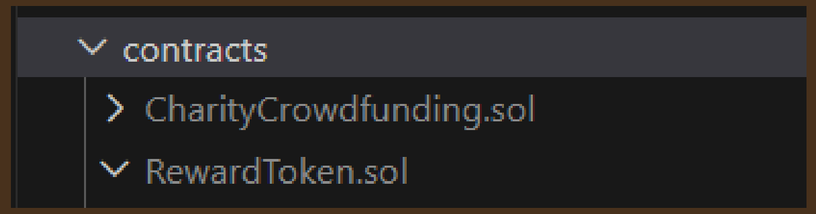
0

Finalize | Refund

Tip: Finalize works only after deadline. Refund works only if campaign failed + finalized + you contributed.

Technologies Used

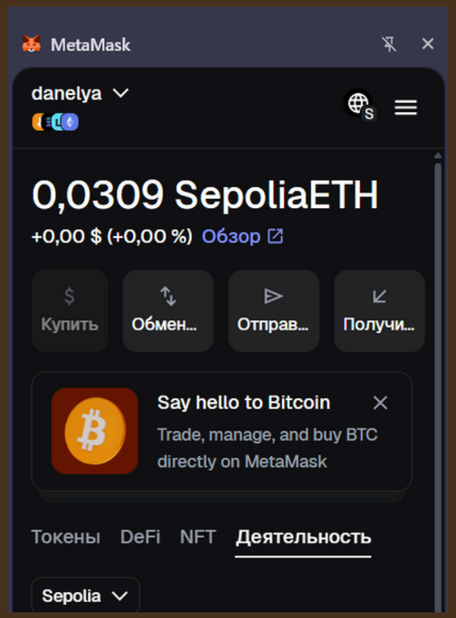
Solidity smart contracts



ERC-20 reward token standard

```
ontracts > RewardToken.sol > RewardToken > setMinter
1   // SPDX-License-Identifier: MIT
2   pragma solidity ^0.8.20;
3
4   import "@openzeppelin/contracts/token/ERC20/ERC20.sol";
5   import "@openzeppelin/contracts/access/Ownable.sol";
6
7   /// Reward token
8   contract RewardToken is ERC20, Ownable {
9
10      // address that is allowed to mint tokens
11      address public minter;
```

Ethereum Sepolia test network (with MetaMask)



Alchemy (RPC connection)

alchemy

Conclusion

This project shows how decentralized technology can improve trust in fundraising systems.
It demonstrates practical blockchain skills such as smart contract development, wallet integration, and DApp interaction.

Thank You for attention!