Proposed zkCrowdfunding Platform Architecture

Overview

The zkCrowdfunding platform revolutionizes traditional crowdfunding by leveraging zero-knowledge proofs (ZKPs) for privacy-focused fundraising. This decentralized platform will allow donors to contribute anonymously, without exposing their identity or contribution amounts, while ensuring transparent verification of fundraising goals.

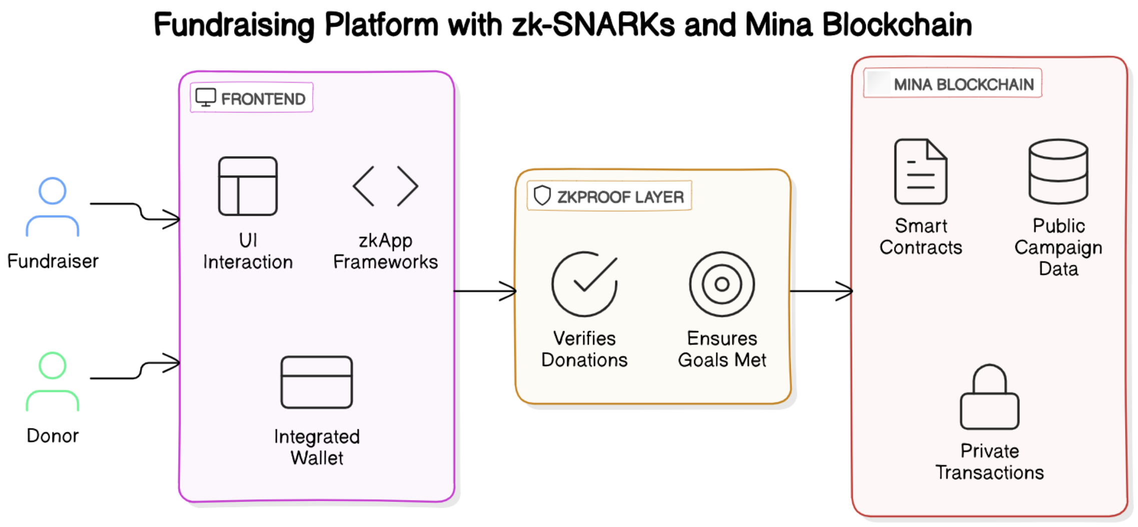
Features

1. **User Roles**:
    - **Fundraisers**: Create campaigns with defined funding goals and deadlines.
    - **Donors**: Contribute anonymously using cryptocurrencies, with their identities and donation amounts protected.
2. **Fundraiser Dashboard Concept**:
    a. **Dashboard Access**: After creating a campaign, the fundraiser can access his dashboard with a given key, and can view a dashboard showing:
        i. **Total Amount Raised** (without showing individual donations).
        ii. **Progress Bar**: How close they are to the funding goal.
        iii. **Time Remaining**: Countdown until the campaign deadline.
    b. **Campaign Expiration**: After the deadline, the dashboard will either:
        i. **Close** (if the goal isn't met), but the fundraiser can still view campaign details.
        ii. **Success Mode** (if the goal is met), where funds can be withdrawn.
    c. **Post-Campaign Access**: Even after the campaign ends, the fundraiser can still view key details like total funds raised, though the dashboard becomes read-only after a set time (e.g., 30 days).
3. **Frontend Interface**:

- Built on Mina Protocol: Using modern web technologies such as HTML, CSS, and JavaScript, with potential integration through Protokit.
- Interaction with Wallets: Donors interact with zkApps through supported wallets, such as Auro Wallet, or any supported wallet, or metamask if we will be using Eth.

4. **Smart Contracts**:
    - The smart contract serving as the backend of the zkCrowdfunding platform shall:
        - Manage campaign creation, enforcing rules and storing campaign data on Mina Protocol's blockchain.
        - Process anonymous donations using zero-knowledge proofs (ZKPs) via ZK-SNARKs/ZK-STARKs.
        - Track cumulative donations and generate ZKPs for fundraising goal validation.
        - Release collected funds to fundraisers upon goal verification via ZKP.
        - Utilize Mina Protocol's blockchain for immutable public record-keeping of campaign milestones.

Workflow



Campaign Creation

1. Fundraiser submits campaign details via frontend UI, including:
    - Funding goal
    - Campaign description
    - Deadline
    - Fundraiser's wallet address (for fund disbursement)
2. Backend creates smart contract on Mina blockchain, storing:
    - Campaign ID
    - Funding goal
    - Deadline
    - Fundraiser's wallet address
3. Smart contract stores details on-chain for immutability and transparency.

Donation Submission

1. Donor contributes via frontend using wallet (Auro, Pallad or Cloriol).

2. Backend generates zero-knowledge proof (ZKP) using zk-SNARKs/zk-STARKs for anonymity.
3. Smart contract verifies ZKP, updates total contributions, and stores aggregate value on-chain.

Ongoing Campaign Monitoring

1. Backend tracks total funds raised and updates progress in real-time.
2. Deadline enforcement is automated.
3. Users can view campaign progress and total donations.

Goal Verification

1. Backend generates ZKP to confirm fundraising goal achievement.
2. Smart contract verifies ZKP on-chain.
3. Campaign marked as successful upon verification.

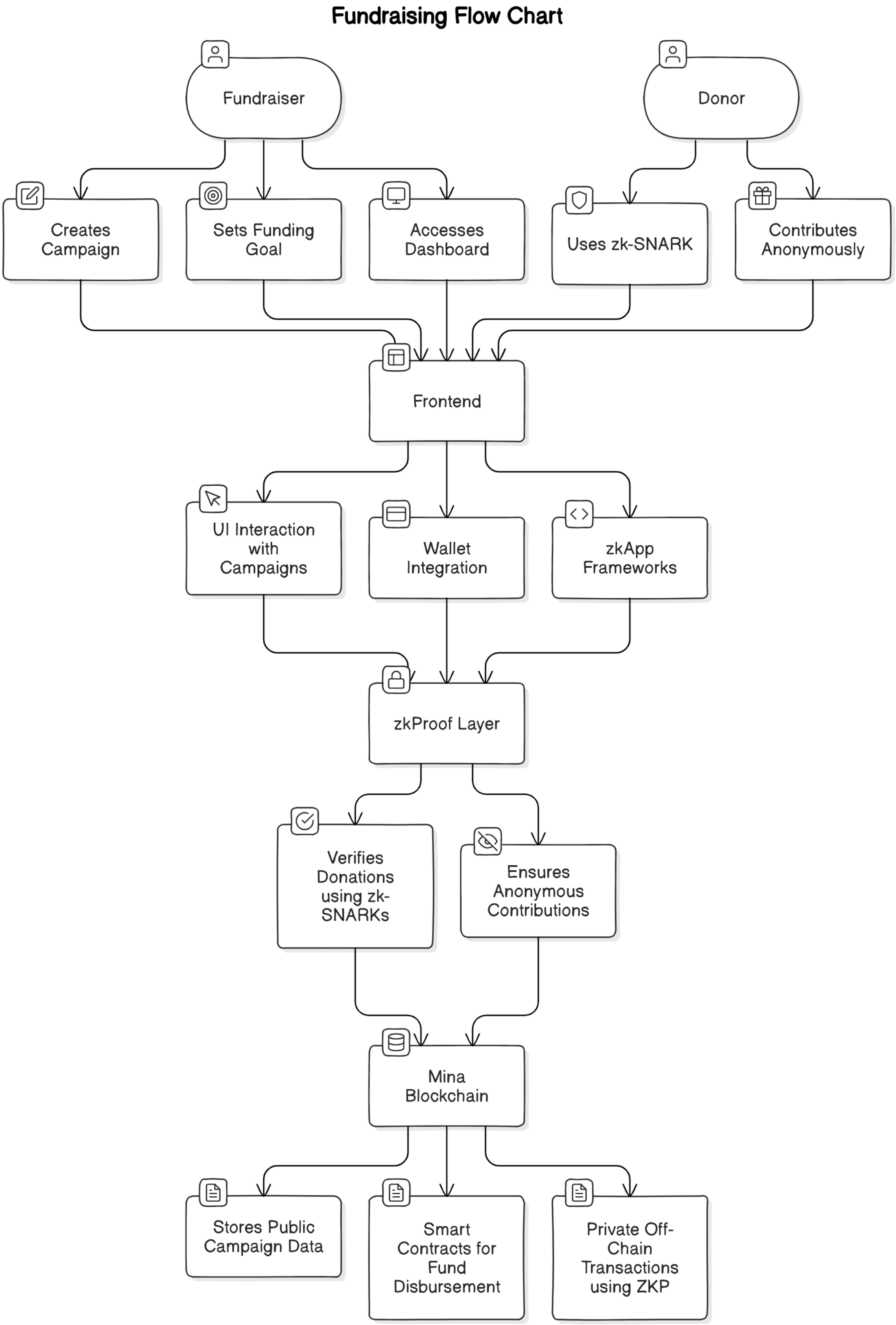
Fund Disbursement

1. Backend initiates fund transfer to fundraiser's wallet address stored in the smart contract.
2. Smart contract ensures secure, automated fund transfer.
3. Transaction recorded on-chain for transparency.

Post-Campaign Analysis and Verification

1. Campaign data publicly verifiable (total funds, goal, deadline).
2. Donor identity remains anonymous.

Flowchart Diagram


Fundraising Flow Chart
Fundraiser
Donor
Creates Campaign
Sets Funding Goal
Accesses Dashboard
Uses zk-SNARK
Contributes Anonymously
Frontend
UI Interaction with Campaigns
Wallet Integration
zkApp Frameworks
zkProof Layer
Verifies Donations using zk-SNARKs
Ensures Anonymous Contributions
Mina Blockchain
Stores Public Campaign Data
Smart Contracts for Fund Disbursement
Private Off-Chain Transactions using ZKP