International Journal for Modern Trends in Science and Technology
Volume 12, Special Issue 01, pages 206-211.
ISSN: 2455-3778 online
Available online at: http://ijmtst.com/ICRCSET2026.html
DOI: https://doi.org/10.5281/zenodo.19536540


Blockchain-Enabled Charity Donation System for Transparency and Trust

Thotakura Meghana, Patan Hazira Bi, Katteda Moulika, Kolla Sailesh Kumar, Inaganti Bhavana

Department of CSE(Data Science), Bapatla Engineering College(Autonomous), Bapatla, Andhra Pradesh, India

To Cite this Article
Thotakura Meghana, Patan Hazira Bi, Katteda Moulika, Kolla Sailesh Kumar & Inaganti Bhavana (2026). Blockchain-Enabled Charity Donation System for Transparency and Trust. International Journal for Modern Trends in Science and Technology, 12(SI01), 206-211. https://doi.org/10.5281/zenodo.19536540

Article Info
Received: 02 March 2026; Revised: 01 April 2026; Accepted: 04 April 2026.

KEYWORDS

Blockchain, Ethereum, Smart Contracts, Charity Donation, DAO Governance, IPFS, Decentralized Application, Transparency, Donor Accountability

ABSTRACT

Traditional philanthropic frameworks often struggle with financial opacity and a reliance on centralized intermediaries, which frequently leads to an erosion of donor trust and systemic mismanagement. This paper proposes a Decentralized Charity Fund Management System that mitigates these risks by encoding the complete donation lifecycle within Ethereum smart contracts, ensuring transparency and accountability by design. Utilizing a governance model inspired by Decentralized Autonomous Organizations (DAOs), the system grants donors proportional voting rights based on their contributions, empowering them to collectively oversee fund disbursement. Capital is released to campaign organizers only after a majority of donors approve specific withdrawal proposals, which must be supported by cryptographic expenditure proofs hosted on the InterPlanetary File System (IPFS). Additionally, the system features an autonomous refund mechanism that activates if a campaign fails to reach its financial target by a set deadline, allowing for the direct reclamation of funds without central intervention. Implementation via a React-based decentralized application (DApp) and validation through Hardhat-based testing confirm that this frameowrk enforces all governance rules deterministically, effectively eliminating the need for centralized authority in the charitable ecosystem.

I. INTRODUCTION

The global philanthropic landscape, which manages trillions in annual capital flow, is currently hindered by a "social trust deficit" fueled by the opaque nature of centralized intermediaries. These traditional platforms often act as informational bottlenecks, where the asymmetry between a donor's contribution and its real-world impact leads to "donor fatigue" and a measurable retreat from charitable participation worldwide. To bridge this gap, this research proposes the Decentralized Charity Fund Management System (DCFMS), a transformative architecture that shifts the

paradigm from "institutional promises" to cryptographic certainty. Built upon the Ethereum blockchain, the DCFMS utilizes Solidity-based smart contracts to function as a deterministic escrow agent, ensuring that funds are governed by immutable code rather than human discretion. By integrating a DAO-inspired governance framework, the system democratizes the oversight process, granting evidence, recording unique cryptographic Content Identifiers (CIDs) on-chain to prevent the manipulation of receipts or project logs. By automating the entire donation lifecycle—from campaign initialization to autonomous refund protocols for failed goals—the DCFMS eliminates the "single point of failure" inherent in centralized non-profit boards. This comprehensive Web3 ecosystem not only optimizes administrative efficiency by reducing overhead costs but also restores the "agency" of the individual donor, providing a borderless, transparent, and security-first solution to the endemic challenges of modern global giving

II. BACKGROUND AND RELATED WORK

A. Blockchain Fundamentals and Smart Contracts

At its core, a blockchain is a decentralized ledger system where data is organized into sequential blocks, each linked to its predecessor via a cryptographic hash. This architecture ensures immutability; modifying a single record would necessitate the re-computation of all subsequent blocks and the consensus of the majority of the network. While Bitcoin introduced this concept for peer-to-peer payments, Ethereum expanded its utility by integrating a programmable layer. By using the Solidity programming language, developers can create smart contracts—self-executing scripts that run across all network nodes. Once these contracts are live, their logic and transaction history are etched permanently into the public record, ensuring high levels of transparency and auditability [1, 2].

B. Decentralized Autonomous Organizations (DAOs)

A Decentralized Autonomous Organization (DAO) represents a shift from traditional, top-down management to a governance model driven by stakeholders. In these systems, organizational decisions are finalized through on-chain voting rather than executive decree. Early iterations typically assigned voting power based on token holdings, but the model has since evolved to support various specialized governance frameworks [3]. In the realm of charitable giving, DAO structures are especially potent. Donors—having already invested capital—possess a vested interest in the organization's success, making them ideal candidates for governance roles where their financial stake encourages ethical and responsible oversight.

C. Decentralized Storage via IPFS

The InterPlanetary File System (IPFS) is a peer-to-peer network designed for decentralized data storage. Unlike the standard web (which uses location-based URLs), IPFS utilizes content-addressing, where files are identified by a unique cryptographic hash of their contents. Because any change to a file results in a completely different hash, it is impossible to secretly alter a document once its identifier is stored on the blockchain. For charitable initiatives, this provides a robust method for storing proof of expenditure. When a receipt or project report is pinned to IPFS and its hash is recorded in a smart contract, the data becomes tamper-proof, ensuring that campaign organizers cannot retroactively modify project documentation [4].

D. Literature Review and Related Work

The intersection of blockchain technology and the non-profit sector has been a growing area of academic interest.

Early Phase: Initial studies focused on basic ledger transparency, allowing donors to observe the flow of funds post-donation without providing any mechanism for intervention.

Intermediate Phase: Later research introduced escrow-based smart contracts that release funds only when specific milestones are met. However, these systems often relied on oracles or central administrators to verify project progress, which creates a "central point of failure" or trust.

The Proposed Gap: This paper builds upon previous work by eliminating the need for a trusted third party. Instead of relying on an external oracle, this system employs a direct donor voting mechanism, empowering the community to collectively validate withdrawal requests and ensure funds are utilized legitimately.

III. SYSTEM ARCHITECTURE

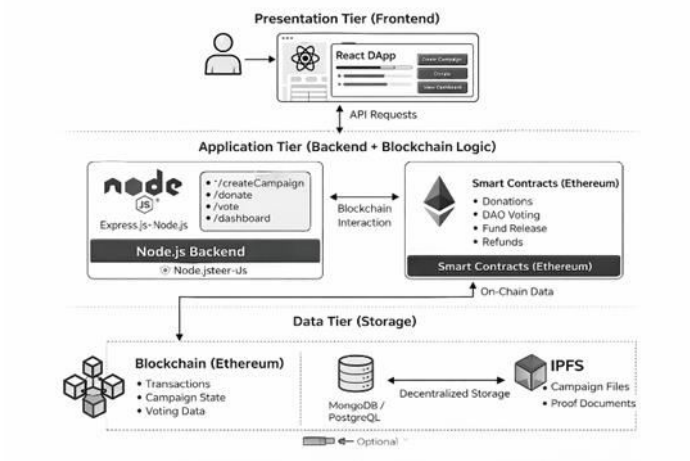

Fig:System Architecture of the framework

A. Structural Framework

The system utilizes a three-layer architecture to balance security and usability. The Blockchain Layer serves as the immutable ledger, hosting the Ethereum smart contract that records all campaign states. TheStorage Layer leverages IPFS for decentralized, tamper-proof hosting of large files like images and reports. Connecting these is the Presentation Layer, a React-based interface that interacts with the blockchain through ethers.js and the MetaMask wallet. The architecture is further enhanced by a Middleware Layer built with Node.js and Express.js. This backend acts as a high-performance indexing service that caches on-chain events—such as campaign creation and successful donations—into a local database. By offloading complex queries (like filtering campaigns by category or popularity) from the Ethereum RPC provider to this centralized middleware, the system achieves significantly lower latency and reduces the computational burden on the client-side application.

B. Campaign Lifecycle Management

Campaigns operate as a Finite State Machine to ensure fund security. Each project starts in the Active state to collect donations. If the goal is met by the deadline, it transitions to GoalReached, enabling fund withdrawal requests. If the goal is missed, it enters the Failed state, while an administrator can trigger a Cancelled state if necessary. Both of these final states automatically lock the funds and activate the donor refund protocol.

C. DAO Governance and Voting

Governance is decentralized, giving donors direct control over fund disbursement. To withdraw money, a creator must submit a proposal backed by IPFS-stored documentation. Donors then vote based on their contribution stake. Once a proposal crosses a predefined threshold—typically 51% approval—the smart contract automatically releases the requested funds, removing the need for a central intermediary or trusted administrator.

D. Security of the Refund Mechanism

The system prioritizes donor protection through an atomic refund process. If a campaign fails or is cancelled, donors can trigger a self-service refund function. The contract follows a "check-reset-transfer" logic, verifying the donor's balance and immediately nullifying their internal record before sending the ETH. This structural design prevents re-entrancy attacks and ensures that funds cannot be claimed more than once.

E. Data Integrity via IPFS Integration

To avoid the high costs of on-chain data storage, the system stores only cryptographic Content Identifiers (CIDs) on the blockchain. These CIDs point to full documents and images hosted on IPFS. Because any modification to a file changes its unique hash, the on-chain CID acts as a permanent seal of authenticity. Users can verify any report or receipt by ensuring its current hash matches the original record stored in the smart contract.

IV. TECHNICAL IMPLEMENTATION

The system is built on a modern decentralized stack, utilizing Solidity 0.8.x for its core smart contracts to leverage native overflow protection and Hardhat for a robust development and testing environment. The user interface is a React 18 application bundled with Vite, interacting with the Ethereum network through the ethers.js v6 library and the MetaMask wallet extension. To ensure high efficiency, the technical stack utilizes content-addressed storage via IPFS, which allows the system to reference large metadata files on-chain without incurring the prohibitive gas costs of storing binary data directly on the Ethereum state.

To optimize contract execution, the CharityDonation logic bifurcates data storage by using compact 128-bit unsigned integers for financial variables and Custom Errors (such as CampaignNotActive or AlreadyVoted) to minimize the data footprint of failed transactions. The

frontend architecture is centered around a Web3Context provider for global state management, which exposes the contract instance through a custom hook to various specialized pages like the AdminPanel and CampaignDetail view. Furthermore, the data-fetching layer employs parallel request processing to simultaneously retrieve financial data and IPFS-linked documentation, significantly reducing perceived latency when interacting with public RPC endpoints.The backend infrastructure is powered by Node.js and the Express.js web framework. This environment manages server-side logic, including user authentication and the synchronization of blockchain events via WebSockets

V. RESULTS AND DISCUSSION

FIG 2

FIG 2,3,4:INPUTS AND OUTPUTS OF OUR CHARITY DONATION SYSTEM

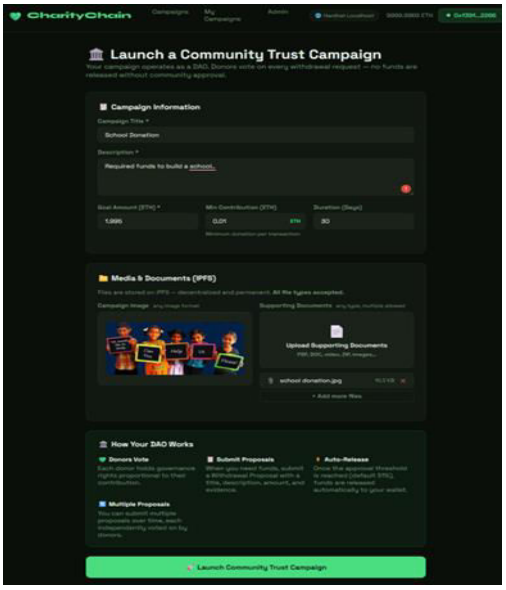

FIG 3

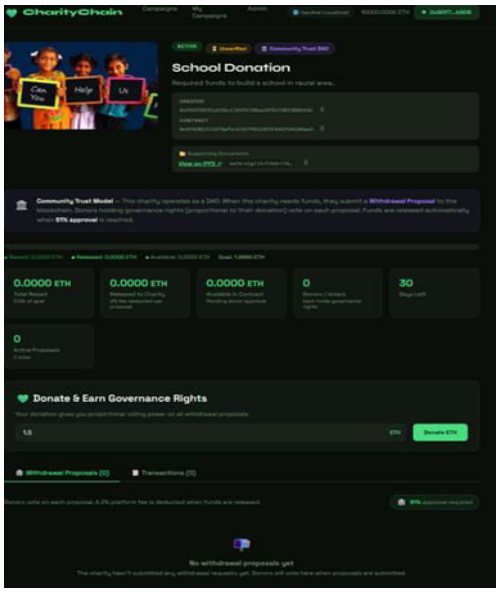

FIG 4

A. Governance Performance

The DAO voting mechanism was benchmarked with a two-donor campaign requiring ceil(2 × 51/100) = 2 approvals for automatic execution. Sequential vote casting was confirmed to maintain the Pending state after the first vote and automatically execute the proposal upon the second vote, transferring funds to the creator without requiring any additional administrative transaction. This demonstrates zero-latency governance execution, a key advantage over off-chain voting systems that require manual result submission.

B. Gas Efficiency

Platform fee calculation is performed using integer arithmetic with a configurable fee percentage (0–10%), avoiding floating-point approximation errors.

C. Comparison with Existing Systems

Compared to centralized charity platforms such as GoFundMe and JustGiving, the BCDS offers complete financial auditability, elimination of platform custody risk, and donor governance rights. Compared to earlier blockchain charity proposals [3][4], the system provides automated fund release (eliminating manual admin steps), IPFS-linked expenditure proofs, multi-wallet support, and a fully functional production-ready frontend. The configurable vote threshold accommodates varying governance requirements without contract redeployment.

VI. SECURITY ANALYSIS

The CharityDonation smart contract is architected to neutralize common blockchain vulnerabilities through industry-standard design patterns. To thwart re-entrancy attacks, the contract strictly follows the Checks-Effects-Interactions model, ensuring all internal state updates occur before any external ETH transfers. Furthermore, the integration of Solidity 0.8.x provides native protection against integer overflows and underflows, eliminating the need for external math libraries. Access control is enforced via lightweight custom error guards, such as NotCreator and NotOwner, which maintain a minimal attack surface by replacing complex on-chain role registries with direct logic checks. To preserve operational integrity, the system implements several functional safeguards. The TooManyActiveProposals constraint prevents Denial-of-Service (DoS) attacks by limiting the number of simultaneous pending requests per campaign. Additionally, the CreatorCannotDonate rule and BelowMinContribution threshold ensure that voting power is distributed fairly among legitimate, economically invested donors rather than being manipulated through self-funding or trivial contributions. While public blockchains are inherently susceptible to front-running, the contract's binary voting system and automated execution logic ensure that outcomes remain deterministic and transparent, regardless of transaction ordering within a block.

VII. CONCLUSION AND FUTURE WORK

This paper has presented the design, implementation, and validation of a Decentralized Charity Fund Management System that uses Ethereum smart contracts, DAO-style governance, and IPFS-based document storage to address the structural weaknesses of conventional charity platforms. The system shifts custody of donated funds from a central administrator to tamper-resistant smart contract code, and shifts decision-making authority from a board of directors to the contributing donor community. Automated test results confirm that all governance invariants are enforced reliably and that no centralized authority is necessary at any stage of the donation lifecycle.

Several directions for future work present themselves. Quadratic voting — where voting power grows as the square root of the stake rather than linearly — could reduce the influence of large donors and produce more equitable governance outcomes. Deploying the contract on Layer-2 Ethereum networks such as Arbitrum or Optimism would reduce transaction costs substantially, making the system viable for small-value campaigns where gas fees currently represent a non-trivial proportion of the donated amount. Integration with decentralized identity protocols would allow the system to verify that campaign creators meet certain credibility criteria without requiring a central verification authority. Finally, a controlled real-world pilot deployment would provide empirical data on donor participation rates and governance engagement that simulation alone cannot capture.

Conflict of interest statement

Authors declare that they do not have any conflict of interest.

REFERENCES

[1] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," Bitcoin.org, Whitepaper, Oct. 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[2] G. Wood, "Ethereum: A Secure Decentralised Generalised Transaction Ledger," Ethereum Project Yellow Paper, vol. 151, pp. 1–32, 2014.

[3] A. Norta, "Establishing Mutually Assured Accountability Using a Decentralized Autonomous Organization for Managing Smart Contracts," in Proc. Int. Conf. Availability, Reliability and Security (ARES), Salzburg, Austria, 2015, pp. 449–456.

[4] J. Benet, "IPFS – Content Addressed, Versioned, P2P File System," arXiv preprint arXiv:1407.3561, Jul. 2014.

[5] V. Buterin, "A Next-Generation Smart Contract and Decentralized Application Platform," Ethereum White Paper, 2014. [Online]. Available: https://ethereum.org/en/whitepaper/

[6] T. Hewa, M. Ylianttila, and M. Liyanage, "Survey on Blockchain Based Smart Contracts: Applications, Opportunities and Challenges," Journal of Network and Computer Applications, vol. 177, p. 102857, Mar. 2021.

[7] Charities Aid Foundation, "CAF World Giving Index 2022," CAF, London, UK, Technical Report, 2022.

[8] K. Bhuptani and S. Moharir, "Blockchain Based Transparent Charity Donation System," in Proc. IEEE Int. Conf. Advances in Computing, Communications and Informatics (ICACCI), Bangalore, India, 2018, pp. 1–6.

[9] K. K. Kommineni, P. Ande, "Blockchain-driven key management and privacy-preserving data Aggregation Scheme for SDN-enabled MANETs," International Journal of Intelligent Engineering and Systems, vol. 18–18, no. 9, pp. 601–615, 2025, doi: 10.22266/ijies2025.1031.39.

[10] N. Szabo, "Smart Contracts: Building Blocks for Digital Markets," Extropy: Journal of Transhumanist Thought, vol. 16, 1996.

[11] OpenZeppelin, "OpenZeppelin Contracts," GitHub Repository, 2023. [Online]. Available: https://github.com/OpenZeppelin/openzeppelin-contracts

[12] A. Narayanan, J. Bonneau, E. Felten, A. Miller, and S. Goldfeder, Bitcoin and Cryptocurrency Technologies. Princeton, NJ: Princeton Univ. Press, 2016.

[13] S. Alzahrani and N. Bulusu, "Block-Supply Chain: A New Anti-Counterfeiting Supply Chain Using NFC and Blockchain," in Proc. 1st Workshop on Cryptocurrencies and Blockchains for Distributed Systems (CryBlock), Munich, Germany, 2018, pp. 30–35