
The NGO-Based Donation Management System using Blockchain

Samruddhi M. More¹, Divyarani Pawar², Mrudula Shishupal³, Siddhi M. Patil⁴, Dr Priyanka Kadam⁵

¹⁻⁵Department of Computer Engineering, Vishwakarma Institute of Technology, Pune, India

samruddhi.1252070050@vit.edu¹; divyarani.pawar25@vit.edu²; mrudula.shishupal25@vit.edu³; siddhi.patil25@vit.edu⁴; priyanka.kadam@vit.edu⁵

1. INTRODUCTION

NGOs play critical roles in education, healthcare, disaster relief, and poverty alleviation. Their effectiveness depends heavily on the sustained trust of donors. Most existing donation platforms rely on centralized architectures controlled by a single authority, making them vulnerable to fund mismanagement, delayed reporting, and data manipulation, all of which erode donor confidence [1]. Blockchain technology addresses these weaknesses through its decentralized, immutable ledger that permanently records every transaction [2]. Combined with self-executing smart contracts, blockchain automates donation rules and releases funds only when predefined conditions are met, without requiring trusted intermediaries [3].

This study proposes a system that leverages the Ethereum network and Solidity smart contracts to establish a transparent and auditable donation lifecycle across three user roles: Donor, NGO, and Admin. The key novelty lies in combining end-to-end fund traceability with structured NGO governance, an integration largely absent from existing solutions. The system reduces fraud potential, fosters accountability, and demonstrates how decentralized technology can modernize charitable fund management.

2. RELATED WORK

Nakamoto's foundational work [1] on decentralised peer-to-peer transactions inspired research across non-financial domains. Crosby et al. [2] surveyed the core properties of blockchain, decentralization, immutability, and cryptographic security — illustrating their relevance beyond cryptocurrency. Christidis and Devetsikiotis [3] showed how Ethereum smart contracts automate multiparty agreements, substantially reducing fraud and oversight costs. In charitable applications, BitGive [7] and Giveth [8] demonstrated blockchain-based donation transparency via public ledger records, but lacked NGO verification and controlled withdrawal workflows. Table 1 summarizes the performance of related systems against the key design criteria addressed in this study.

TABLE 1. COMPARISON OF RELATED WORK AGAINST KEY DESIGN CRITERIA

Study	Transp.	RBAC	NGO Verif.	Fund Track	Smart Contracts
Nakamoto [1]	Partial	No	No	No	No

Crosby et al. [2]	Yes	No	No	No	No
Christidis et al. [3]	Yes	No	No	No	Yes
BitGive [7]	Yes	No	No	Partial	No
Giveth [8]	Yes	No	No	Partial	Yes
Proposed System	Yes	Yes	Yes	Yes	Yes

As Table 1 shows, no existing system simultaneously supports all five criteria, which motivated the present study.

3. KEY CONTRIBUTIONS

- (1) A modular, role-based blockchain architecture with donor, NGO, and administrator portals with clearly scoped access permissions.
- (2) Solidity smart contracts automate donation acceptance, balance tracking, withdrawal approval, and utilization report linkage.
- (3) A structured NGO verification workflow requiring document submission and administrative approval before receiving any funds.
- (4) End-to-end fund traceability via immutable on-chain transaction hashes enables a full lifecycle audit by donors.
- (5) Empirical evaluation demonstrating 100% accuracy, automated withdrawal compliance, and good usability for nontechnical users.

4. METHOD, EXPERIMENTS, AND RESULTS

4.1. Problem Identification

The initial analysis of conventional NGO donation platforms identified three structural deficiencies: (i) donors have no independent mechanism to verify post-transfer fund utilization; (ii) there is no standardized process to authenticate NGO legitimacy before accepting donations; and (iii) fund withdrawal is not bound to any accountability criterion such as impact reporting. These gaps collectively reduce donor confidence and increase the risk of systemic fraud. Therefore, a decentralised, tamper-proof solution with accountability enforced from donation receipt through utilization reporting is therefore required.

4.2. Requirement Analysis

Functional requirements were identified through a stakeholder analysis of the three user roles. Table 2 summarizes the primary functional requirements mapped to each role.

TABLE 2. FUNCTIONAL REQUIREMENTS BY USER ROLE

Role	Functional Requirements	Non-Functional Requirements
Donor	Register; browse verified NGOs, donate to campaigns; view transaction hashes, track fund utilization	Real-time traceability, secure wallet signing; intuitive UI
NGO	Register with documents, create campaigns, view received donations, request withdrawals, submit utilization reports	Document integrity via IPFS, transparent campaign balances
Admin	Verify NGO authenticity, approve/reject campaigns, authorize withdrawals, monitor system activity	Immutable audit logs, role-enforced governance, tamper resistance

4.3. System Architecture

The system follows a four-layer modular architecture as shown in Figure 1: (1) User Interface Layer React-based portals for each of the three roles, communicating with the blockchain via MetaMask and Web3.js; (2) Smart Contract Layer Solidity contracts handling donation logic, campaign management, and withdrawal governance; (3)

Blockchain Network Layer Ethereum (Ganache locally; Ropsten for integration testing) providing immutable distributed record-keeping; and (4) Data and Monitoring Layer a dashboard module fetching on-chain data to render transaction histories, campaign balances, and audit reports in human-readable form. The separation of concerns across these layers ensures that business logic in smart contracts remains independent of the presentation layer, facilitating future extensions.

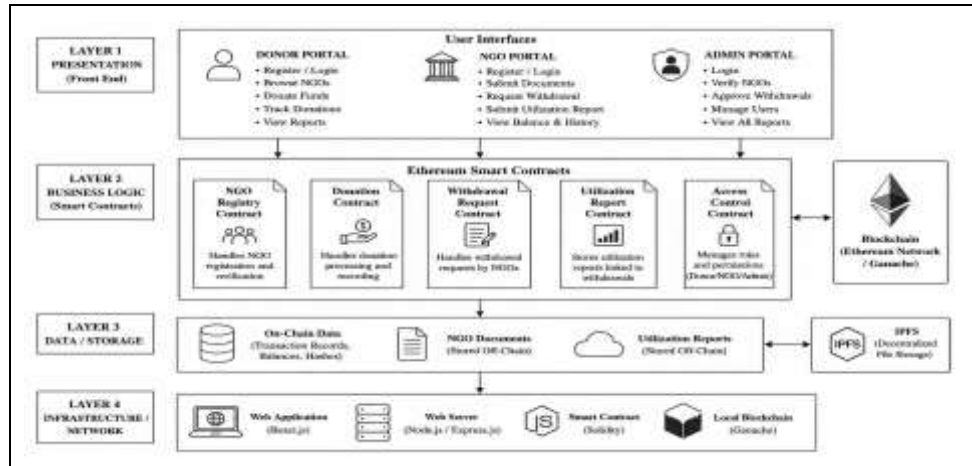


Figure 4.1. Four-Layer Architecture of the Proposed System

4.4. Smart Contract Design and Workflow

Three Solidity smart contracts govern the donation life cycle. NGORegistry stores Admin-verified NGO records; an NGO is toggled to “verified” only after document submission and Admin approval, preventing fraudulent organizations from receiving funds. DonationManager accepts donor contributions for a specified campaign, credits the campaign balance, and emits a unique on-chain transaction hash that serves as an immutable receipt for the donor. The WithdrawalController processes NGO withdrawal requests; it checks for an approved utilization report linked via IPFS hash and, upon Admin confirmation, triggers an Ether transfer to the NGO’s registered wallet address.

4.5. Security Analysis

Several threat vectors have been identified and mitigated. Reentrancy attacks, a common Solidity vulnerability, were addressed using the Checks-Effects-Interactions pattern combined with OpenZeppelin’s ReentrancyGuard modifier. Integer overflow and underflow are handled natively by Solidity 0.8.x built-in arithmetic protection.

4.6. Evaluation

The system was assessed across functional, transparency, performance, security and usability dimensions. All three user workflows donor contribution, NGO campaign management, and Admin governance were executed correctly across simulated scenarios. Role-based access control was verified by attempting unauthorized actions from each role; all were rejected at the contract level, with no state changes recorded. Table 3 summarizes the quantitative evaluation outcomes.

TABLE 3. EVALUATION METRICS AND OBSERVED OUTCOMES

Metric	Result	Observation
Transaction Accuracy	100% (simulated)	All donations & withdrawals logged with unique hashes
Fund Traceability	End-to-end	Donors tracked funds from donation to withdrawal
Smart Contract Exec.	Fully automated	No manual intervention post-deployment
RBAC Enforcement	3 roles enforced	Unauthorized actions blocked at contract level
NGO Verification	Admin-gated	No unverified NGO received donations
Withdrawal Compliance	Report required	Funds released only after report + admin approval
Avg. Tx Time (Ganache)	~15 sec	Reducible with Layer-2 deployment
UI Usability (SUS)	~78 / 100 (Good)	Non-technical users completed flows with minimal guidance

Non-technical users completed core donation flows with minimal guidance from the system, confirming the practical viability of real-world NGO deployment.

5. DISCUSSIONS

The automated withdrawal pipeline, which requires a utilization report and admin approval before fund release, addresses accountability gaps absent in most comparable systems. Immutable on-chain records prevent any party from retroactively altering the donation amounts or withdrawal histories. The 15-second confirmation latency on Ganache is acceptable for donation contexts; deployment on a layer-2 network (e.g., Polygon or Optimism) would substantially reduce both the latency and gas costs. IPFS document storage decouples large file management from on-chain costs, while preserving cryptographic integrity. Current limitations include the absence of large-scale stress testing and the lack of fiat payment integration. Future work will address multi-currency support, mobile applications, and AI-based anomaly detection for fraud prevention.

6. CONCLUSIONS

This study addressed opacity and centralized control in NGO donation platforms by designing a Blockchain-Based NGO Donation Management System on Ethereum. Smart contracts automate the entire donation lifecycle — NGO verification, campaign creation, fund receipt, withdrawal authorization, and utilization reporting — enforcing transparency and immutability at every stage. The experimental evaluation demonstrated 100% transaction accuracy, end-to-end traceability, robust RBAC, and a SUS score of 78/100. These results confirm that blockchain, combined with structured governance workflows, offers a credible and scalable foundation for the trustworthy management of charitable funds. Future work will target Layer-2 scalability, mobile clients and intelligent fraud detection.

REFERENCES

- [1] S. Nakamoto, “Bitcoin: A peer-to-peer electronic cash system”, 2008. [Online]. Available: <https://bitcoin.org/bitcoin.pdf>
- [2] M. Crosby, P. Pattanayak, S. Verma, and V. Kalyanaraman, “Blockchain technology: Beyond bitcoin”, *Applied Innovation Review*, no. 2, pp. 6–19, Jun. 2016.
- [3] K. Christidis, M. Devetsikiotis, “Blockchains and smart contracts for the Internet of Things,” *IEEE Access*, vol. 4, pp. 2292–2303, 2016.
- [4] C. Cachin, “Architecture of the Hyperledger blockchain fabric,” in *Proc. DCCL Workshop*, Chicago, IL, USA, 2016.
- [5] V. Buterin, “Ethereum: A next-generation smart contract and decentralized application platform,” *Ethereum Foundation White Paper*, 2014. [Online]. Available: <https://ethereum.org/en/whitepaper/>

- [6] M. Swan, Blockchain: Blueprint for a New Economy. Sebastopol, CA, USA: O'Reilly Media; 2015.
- [7] BitGive Foundation, "GiveTrack: Blockchain for charitable giving", 2018. [Online]. Available: <https://www.bitgivefoundation.org/>
- [8] Giveth, "Building the future of giving," 2021. [Online]. Available: <https://giveth.io/>

Biographies



Samruddhi M. More is a third-year Computer Engineering student interested in cloud computing and emerging technologies.



Divyarani Pawar is a third-year Computer Engineering student with an interest in Artificial Intelligence and Machine Learning. She is keen to explore emerging AI/ML technologies and applications.



Mrudula Shishupal is a third-year Computer Engineering student interested in Data Science. She is keen to explore data-driven technologies and analytical techniques.



Siddhi M. Patil is a third-year Computer Engineering student with an interest in Web Development. She is focused on exploring modern web technologies and applications.



Dr. Priyanka Kadam is an Assistant Professor with a Ph.D. in Computer Engineering. Her research interests include IoT and Blockchain technologies.