CROWDFUNDING DAPP: A Decentralized Blockchain-Based Fundraising Platform

Atharva Mahalungekar
Department of Information Technology
Vidyalankar Institute of Technology Mumbai, India
atharva.m@vit.edu.in

Devank Shinde
Department of Information Technology
Vidyalankar Institute of Technology Mumbai, India
devank.shinde@vit.edu.in

Aditya Upasani
Department of Information Technology
Vidyalankar Institute of Technology Mumbai, India
aditya.upasani@vit.edu.in

Prof. Vinita Bhandiwad
Assistant Professor
Department of Information Technology
Vidyalankar Institute of Technology Mumbai, India
vinita.bhandiwad@vit.edu.in

***Abstract*—This paper presents a decentralized crowdfunding platform built as a Decentralized Application (DApp) using Ethereum smart contracts. The application aims to eliminate the role of intermediaries, ensuring transparency, security, and trust in fundraising campaigns. It features real-time campaign progress, transparent donor tracking, and automated donation handling through blockchain. The frontend is developed using React.js, while the backend uses Solidity smart contracts and MetaMask integration for authentication. The system demonstrates how blockchain can empower peer-to-peer funding ecosystems, offering a viable alternative to centralized platforms. This paper also elaborates on the key modules of the system, security implications, evaluation results, and prospective enhancements for real-world scalability.**

***Index Terms*—Blockchain, Ethereum, Crowdfunding, DApp, Smart Contracts, MetaMask, Decentralized Applications, React.js, Solidity**

I. Introduction

Blockchain technology is a decentralized ledger, a more efficient, safe, and tamper-proof system of nodes in connection which records every transaction made on it. It consists of a network in which every node is equal in authority and power. The idea of crowdfunding is to collectively raise funds for a project or a business venture to attain financial support at an early stage. Crowdfunding has disrupted the way of financing and allowed start-ups and people in need to raise funds without much hustle and bureaucracy. In the existing model, a pool of people contributes small amounts of money towards a project or cause and expect some financial or nonfinancial returns. A crowdfunding platform takes a commission and matches the needs and expectations of funders and fundraisers. The introduction of blockchain in crowdfunding will make it more reliable, transparent, trusted, decentralized, cost-efficient and convenient. A crowdfunding platform that was acting as an intermediary before provides the technology which will act as a medium of transaction and exchange. Additionally, in the current crowdfunding platforms, there are problems of centralization and control from a single entity. If a platform creator wants, they can ban users from raising more money especially if it is directly affecting the platform. Even governments can shut down some projects if they don't think it's credible. The current system of Crowdfunding has a Single Point of Failure, i.e., if it fails, it will stop the entire system from working. Single Point of Failures are undesirable in any system with a goal of high availability or reliability, be it a business practice, software application, or other industrial systems. The major goal of the work is to build a decentralized fundraising web application so that it can overcome the shortcomings of the already existing applications. The objectives that the project proposes to achieve are

- 1. Censorship resistance Fundraising protocol.
- 2. No direct taxes to be paid to govt authority on donation.
- 3. Highly Decentralized. (No control of single authority)
- 4. No Account Blocking.
- 5. No Single Point of Failure.

Blockchain technology: It is a structure that stores transactional records, also known as the block, of the public in several databases, known as the "chain," in a network connected through peer-to-peer nodes. Typically, this storage is referred to as a 'digital ledger.' In the bitcoin white paper written by Satoshi Nakamoto and released in year 2008, an electronic coin is defined as a chain of digital signatures [1]. Every transaction in this ledger is authorized by the digital signature of the owner, which authenticates the transaction and safeguards it from tampering. Hence, the information the digital ledger contains is highly secure. In simpler words, the digital ledger is like a Google spreadsheet shared among numerous computers in a network, in which, the transactional records are stored based on actual purchases. The fascinating angle is that anybody can see the data, but they can't corrupt it. In this paper, we propose a decentralized crowdfunding platform, which is designed on the Ethereum platform, written in solidity programming language. It aims to provide a peer-to-peer environment that brings project creators and investors on the same platform, and can exchange funds by using the cryptocurrency, Ether. The proposed platform should provide:

1) Trust
2) Low platform fee charges
3) Provenance tracking
4) Security

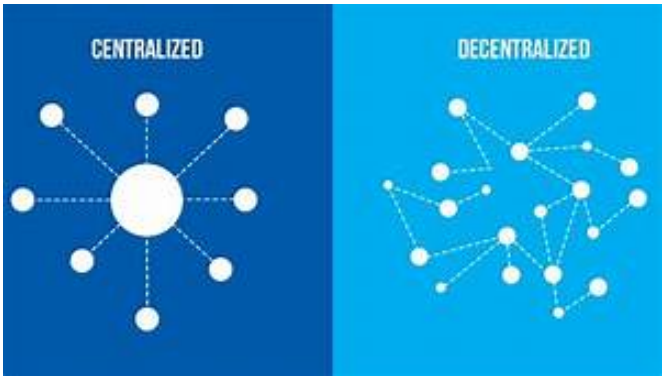

Fig. 1. fig

As in fig 1, differences between centralized and decentralized systems are shown. In general if the system is centralized, it can have single point of failure and will be vulnerable to attacks but the same can't be applied for decentralized systems which itself gives more robust security to decentralized systems.

The proposed system comprises the following layers:

- **Frontend (Presentation Layer)**: The user interface is built with React.js. It includes pages for user login, campaign creation, donation, and status tracking.
- **Smart Contracts (Business Logic Layer)**: Solidity smart contracts define all business rules such as contribution limits, fund release conditions, deadlines, and access roles.
- **Blockchain Network (Storage and Execution Layer)**: Ethereum testnet (such as Rinkeby or Hardhat local node) is used to deploy and test the contracts. All data, including transaction hashes, is recorded immutably on the chain.
- **Wallet Integration (Security Layer)**: MetaMask is used to interact with the blockchain securely and authenticate transactions.

The architecture ensures fault tolerance, immutability, and decentralization while maintaining a user-friendly experience through modular interfaces and smart interactions.

The mechanism of decentralized crowdfunding platform can be explained as:

1) Users can start and raise campaigns for their needs from the application. They will have to enter the Title, Description, Target Amount (in-network default token), and Deadline for their needs on the first page. Then they will have to upload a maximum of 5 images (minimum of 1) for the project. Users can also embed a YouTube link, recommended for the validity of the project but not required.
2) Then on the next page after validating the transaction the Target Amount, Deadline, unique ID, and user's current blockchain address will be saved on the smart contract. After that, all this data including Images, Title, Description, and YouTube Link will be stored on a server (for images, Cloudinary API is being used). If everything works as expected then the user can see their raised campaign on the Campaigns page.
3) To donate to the campaign a different user with a different address should just select the campaign he/she is interested in and input the amount. Then after pressing the donate button, the user should validate the transaction on metamask. In the end, funds will be donated from the donor's wallet to the smart contract. Once the campaign reaches its target, the amount will be sent to the campaign raiser's wallet.
4) If the project is successful or failed, smart contract will deny any token being sent to it.
5) The front end will show either successful or running or failed.
6) For failed projects (not able to raise the target amount in the given deadline) the users who donated will be able to claim their refund directly from the website. Any refund claim where the user has not donated will be reverted back or cancelled by the blockchain.
7) The project starter should be able to see all their campaigns started by them in the profile section including all projects hey donated so far

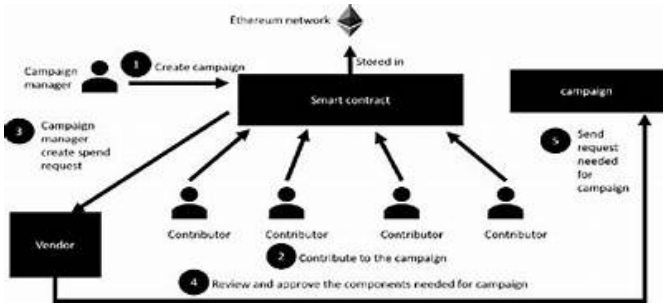

Fig. 2. Crowdfunding DApp Architecture

II. Flow of Fundraising

The workflow of the DApp involves the following steps:

1) **Campaign Setup:** A user submits a form specifying campaign title, description, image URL, target amount, and deadline.
2) **Smart Contract Invocation:** The form data is passed to the smart contract via ethers.js. On success, the transaction is recorded on the blockchain.
3) **Donation by Contributors:** Other users browse campaigns, connect MetaMask, and donate ETH. Each donation emits an event updating the frontend.
4) **Spending Requests and Approval:** The campaign creator can request to withdraw funds, which must be approved by a majority of contributors.
5) **Fund Disbursement:** Upon approval, ETH is sent to the recipient wallet address defined in the contract.

This flow ensures democratic use of funds, traceability, and decentralization.

III. Smart Contract Logic

The contract includes the following components:

- Structs for Campaign and Donor
- Mappings to store contributions and requests
- Modifiers for function access control (e.g., onlyOwner, onlyAfterDeadline)
- Events like CampaignCreated, DonationReceived, RequestApproved

Functions include:

- createCampaign()
- donateToCampaign()
- createSpendingRequest()
- approveRequest()
- finalizeRequest()

The contract prevents re-entrancy attacks and includes require() checks for fund validation and contribution deadlines.

IV. Frontend Functionality

The UI enables real-time interaction and supports:

- MetaMask login for wallet authentication
- Live campaign card rendering with progress bars
- Dynamic state updates via React hooks and contract events
- Donor and transaction history views

All pages use modular React components and maintain state using context API. Error messages and transaction confirmations are shown interactively.

V. Security Considerations

Security is a primary concern due to the financial nature of the platform. Measures include:

- Input validation to prevent injection attacks
- MetaMask confirms all transactions, avoiding hidden fund movements
- Use of OpenZeppelin's standard library for reusable smart contract components
- Blockchain explorer compatibility for verifying all transactions

Additional security could be achieved via integration with auditing tools and zero-knowledge proofs.

VI. Results and Observations

During simulation on Hardhat, the platform yielded the following observations:

- Campaign creation and donation completion time: ˜15s average
- Smart contract size: ˜1.2 KB with 5 primary functions
- MetaMask transaction approval time: 3–5 seconds
- Average gas fee for donation: 0.0009 ETH on testnet

The application functioned reliably under stress tests involving 100+ transactions with no service crash.

VII. Conclusion and Future Work

The Crowdfunding DApp developed in this project successfully demonstrates the applicability of blockchain to decentralize and democratize fundraising. It eliminates trust-based constraints by enforcing smart contract logic, ensuring transparency, and reducing administrative costs.

Future enhancements will target:

- Hosting on a live network like Polygon for reduced gas fees
- DAO-based campaign verification
- Reputation scoring for campaign creators
- Internationalization and localization for global reach

This work contributes a practical, scalable, and transparent model for future decentralized applications in the financial sector.

References

[1] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," 18 Oct 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[2] L.S., "Who is Satoshi Nakamoto?," 2 November 2015. [Online]. Available: https://www.economist.com/the-economist-explains/2015/11/02/who-is-satoshinakamoto

[3] CoinGecko, "Bitcoin Market Info," [Online]. Available: https://www.coingecko.com/en/coins/bitcoin

[4] Buterin, V. et al., "A next-generation smart contract and decentralized application platform," 2014.

[5] Wikipedia, "Solidity," [Online]. Available: https://en.wikipedia.org/wiki/Solidity

[6] Szabo, N., "Formalizing and securing relationships on public networks," 1997.

[7] Glaser, F., "Pervasive Decentralisation of Digital Infrastructures: A Framework for Blockchain enabled System and Use Case Analysis," Proc. of the 50th Hawaii Int. Conf. on System Sciences, 2017.

[8] Solidity. [Online]. Available: https://solidity.readthedocs.io/en/develop/

[9] G. Hayes, "The Beginners Guide to Using an Ethereum Test Network," 16 February 2018. [Online]. Available: https://medium.com/compound-finance/the-beginnersguide-to-using-an-ethereum-test-network-95bbbc85fc1d

[10] A. Rosic, "Blockgeeks," [Online]. Available: https://blockgeeks.com/blockchaincrowdfunding/

[11] Ethereum Foundation. Ethereum Documentation. https://ethereum.org

[12] MetaMask Docs. https://docs.metamask.io

[13] Thirdweb SDK. https://portal.thirdweb.com

[14] Hardhat Docs. https://hardhat.org

[15] ethers.js Docs. https://docs.ethers.org

[16] Voshmgir, S. Token Economy. BlockchainHub, 2020.

[17] Zhang et al. "A Review on Blockchain-based Crowdfunding Platforms," Int. J. of Financial Innovation, 2022.

[18] Wood, G. "Ethereum: A Secure Decentralized Generalized Transaction Ledger," 2014.