Development Of a Crowdfunding Application Powered by Ethereum Blockchain

GEETIKA JHANJI[1], VIDUSHI TYAGI[2], AADITYA GAUR[3], YOGESH SHARMA[4]

[1, 2, 3] *Student, Department of Computer Science and Engineering, Maharaja Agrasen Institute of Technology, Affiliated to G.G.S.I.P. University*

[4] *Assistant Professor, Department of Computer Science and Engineering, Maharaja Agrasen Institute of Technology, Affiliated to G.G.S.I.P. University*

Abstract — Distributed ledger technologies are in path to disrupt economic interactions across markets. The aim of this project is to identify inefficiencies of the current crowdfunding market and address them by combining smart contracts with smart property in a trustworthy decentralized application built on Ethereum. The focus is made on services provided by crowdfunding intermediaries in order to analyze their implementation on blockchains and to design a decentralized application. This methodology points out the potential of crowdfunding decentralized applications to lower market inefficiencies by bypassing third parties and easing trade on secondary markets.

Indexed Terms — Blockchain, Campaign, Crowdfunding, Hashing, Request-Approval, Smart Contracts

I. INTRODUCTION

The practice of funding a project or some kind of venture by raising money from lots of people, especially using the internet, is known as crowdfunding [5]. In today's world where all of us are connected through the power of the internet, crowdfunding is one of the most creative and helpful innovations of our time. It helps people gather funds for various number of reasons, ranging from natural disasters like floods and earthquakes to setting up local businesses or startups. It brings communities together for a common cause.

Usually, crowdfunding apps have limited security measures, so investors risk their money by supporting startups. Fraudulent actions like false claims about reached milestones or even released products, and in some cases, lead to substantial monetary losses. As soon as a startup gets needed financing, it can disappear from the platform with all its creators' funds. Our motivation behind creating this project is to build more transparent communication between investors and startups to avoid such fraudulent acts [2]. In simple terms, we aim to create a secure platform where funds can be raised without the fear of any fraud or malpractice. This is where Blockchain comes in handy. Blockchain is a decentralized distributed ledger that is used to record all the transactions across many systems so that the information cannot be altered subsequently [4].

This paper is structured as follows: Section II provides details on Blockchain. Section III explains the process of Hashing. In Section IV the details about smart contracts have been provided; Section V explains the methodology followed by conclusion and references.

II. LITERATURE SURVEY

A. Crowdfunding and Venture Capital: Substitutes or Complements?

The authors in this paper, basically attempt to study the dynamics of crowdfunding and if venture capitalists and the investors of crowdfunding are complimentary in nature [20].

B. Lemmings in the Crowd: Success and failure of crowdfunding platforms

In this paper, the author tries to analyze the determinants which make a business a success or a failure, especially the crowdfunding platforms [21].

C. Crowdfunding Smart Contract Security and Challenges

In this paper, the author walks us through the different security and challenges that we might come across during and after the implementation of Crowdfunding platform using Blockchain [22].

D. How Blockchain is Revolutionizing Crowdfunding

In this paper, the author explains the limitations of crowdfunding platforms and the benefits of blockchain technology and how it is the future of crowdfunding owing to the ease and transparency of this model [23].

E. Implementation of a Crowdfunding Decentralized Application on Ethereum Master Thesis

In this thesis, the author identifies the inefficiencies of the crowdfunding market and addresses them by combining smart contracts with smart property in a trustless decentralized application built on Ethereum. The focus is made on services pro- vided by crowdfunding intermediaries to analyze their implementation on blockchains and to design a decentralized application [24].

F. Blockchain-Based Crowdfunding

In this paper, the authors try to explain how a crowdfunding platform which was acting as an intermediary before will only provide the technology and name is its own crypto currency which will act as a medium of transaction and exchange. Fundraisers will generate their own currency and everyone on the network will be notified about the project. Funders will buy this crypto currency to claim its share in the project and can withdraw any time by selling the currency and losing the share in a project or transferring it to another project [25].

III. BLOCKCHAIN

A distributed database or ledger that is shared among the nodes of a computer network is known as Blockchain. Blockchain works as a database that stores information and tracks transactions electronically in a digital format. Blockchain maintains a secure and decentralized track record of transactions. The specialty of Blockchain is that it is secure and maintains fidelity throughout the record. Due to this specialty, it is easier to entrust sensitive information through Blockchain Technology.

Information in Blockchain is stored together in groups known as Blocks. Blocks are structures that have a set of information stored inside them. Blocks have storage features that allow them to be linked to the previous block, thereby forming a chain of Blocks, hence the name Blockchain [3]

IV. NEED OF BLOCKCHAIN

In our project, we make major use of Blockchain to overcome the challenges in the current system. Some of the reasons for using Blockchain are as follows: -

A. Decentralization

The decentralization aspect of blockchain makes it one of the hottest technologies to use currently in the tech world because it helps achieve greater and fairer service by transferring control from a centralized authority to a distributed network [1]. Decentralized networks aim to increase the level of trust that participants must place in one another and reduce their ability to establish dominance over one another in ways that may degrade the functionality of the network.

B. Reduced Fees

Most traditional crowdfunding platforms take a significant part of funds that campaign creators raise during their campaign. But Blockchain never engages with intermediaries in financial transactions, so it makes crowdfunding much more economical for creators [8].

C. Transparency

Traditional crowdfunding apps have weak security measures, so contributors usually risk their money by supporting campaigns. These fraudulent actions lead to substantial monetary losses. As soon as a startup gets needed financing, it can disappear from the platform with all its creators’ funds. Smart contracts help people identify both sides of the transaction, so there is a lesser probability of fraud. The technology builds up a more transparent communication between campaign creators and contributors, so blockchain-enabled crowdfunding projects have higher efficiency [7].

D. Speed

With rare exceptions, even those crypto transactions which are slow can be considered fast as compared to other traditional payment methods which may take hours or even days because most blockchain transactions are completed in minutes.

V. HASHING

The process of sending an input string of variable length to a mathematical function known as hash function, thereby resulting in an output of static length is known as hashing [9]. Most cryptocurrencies use a hashing algorithm. For example, in the case of Ethereum, it uses Keccak-256 in a consensus engine called Ethash [6]. A hash function is only cryptographically secure if it has the following properties: -

A. Large Output Space

Hash collision can be found via a brute force search, for which checking as many inputs as the hash function has possible outputs is required.

B. Deterministic

What being deterministic means here is that, for any given input a hash function should always give the same result.

C. Preimage Resistance

For a hash value y, it is difficult to find a message x such that y = hash(k, x)

D. Collision Resistant

If there are two inputs X and Y, it is tough to find a hash value such that hash(k, X) = hash(k, Y) where k is the key value.

VI. SMART CONTRACT

A smart contract automates the execution of an agreement between buyer and seller by writing everything directly into lines of code. Therefore, it helps in informing all the participants about the outcome without the involvement of a third-party [10]. They also execute a workflow, by executing the next set of instruction when some conditions are met.

Smart contracts work by using “if/when…then…” conditions that are written into code on a Blockchain. A network of computer systems carries out the actions when these predetermined conditions have been verified and met. These actions mostly include releasing funds to the appropriate parties, registering a vehicle, sending notifications, or issuing a ticket. The Blockchain is then updated when the transaction is completed. That means the transaction cannot be changed, and only parties who have been granted permission can see the results.

VII. METHODOLOGY

A. Tools and Technology Used

For writing smart contracts, we have used *Solidity* language [14]. To carry out different transactions, every client needs to have a cryptocurrency wallet like *Metamask* [11] installed. For the Ethereum development environment, we have used *Hardhat* [15] which will allow us to compile and run our smart contracts on a development network. *Ethers.js* library and assertion library *Chai* are used with the testing framework to make our smart contract tests easy to write and read. As for the frontend, we have used *Next.js* [12] which is a framework built on top of React. The application state is managed and centralized with the help of Redux [18]. The Web3.js library [13] is used alongside Next.js in front-end code for creating a web3 client. The user interface is controlled using *Tailwind.css* [17].

B. Architecture

Fig. 1. represents the architecture of our Crowdfunding application. This shows how our web application with Solidity as our backend works. All the smart contracts that interact with the Ethereum blockchain are written in Solidity language [14]. We have also used Hardhat [15], which is an Ethereum development environment, along with the Chai assertion library to perform various tests on our smart contracts.

We are using Next.js [12] as our frontend, a framework built on top of React, to serve our JavaScript content to the browser. When a user performs a transaction, they do not reach the server, instead the Ethereum application running inside the web browser uses web3 and communicates with Metamask [11], then Metamask creates a transaction and signs it with the user's private key and sends that transaction to the Ethereum network. These transactions can be tracked at Etherescan [16] to provide transparency of the whole process. These public and private keys will never be sent to the server because you can never ask for your user's private keys. So, here the client has more power and privacy.

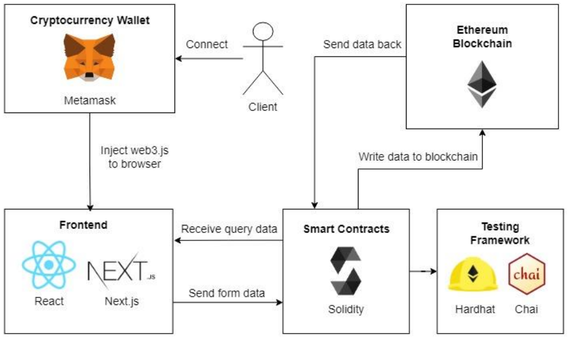


Fig -1: System Architecture of Crowdfunding Application

C. Proposed System

An Ethereum based smart contract is a cryptographic box which stores information, processes inputs, writes outputs and is only accessible to the outside if certain predefined conditions are met and the contracts in Ethereum are written in a special language called solidity. In practice, Ethereum allows for an easy implementation of such smart contracts and in addition Ethereum offers developers online compilers of solidity code. Smart contracts are written in such a way that the entire amount funded by the contributors will safely be kept in smart contracts so that no one can modify it or steal it. The amount will not be given directly to the campaign creator rather it will be held in the smart contract itself.

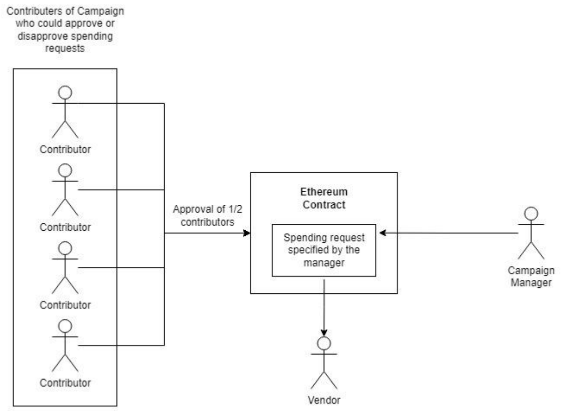

Contributers of Campaign who could approve or disapprove spending requests
Contributor
Contributor
Contributor
Contributor
Approval of 1/2 contributors
Ethereum Contract
Spending request specified by the manager
Campaign Manager
Vendor

Fig -2: High Level Overview of System Implementation

The following are the 6 modules that our application takes care of:

1. *Wallet Connect:* It is required for a user to have a cryptocurrency wallet (e.g., Metamask) to interact with our application. Initially, when a user visits our decentralized crowdfunding application, their cryptocurrency wallet is connected, after which the user can perform various transactions.
2. *Campaign Creation:* A user can create a campaign by providing necessary details such as campaign title, campaign description, target contribution amount, minimum contribution amount and deadline. A small amount of gas fee is charged for every transaction that takes place. Therefore, for every change that needs to happen in the blockchain, we need to provide some amount of money to make that transaction a valid one. After a couple of seconds, when the transaction is completed, a new block is added to the Ethereum blockchain with the contract address. The Home Page then also displays this newly created campaign, with which the user as well as the contributor can interact.
3. *Contributions:* Donors can browse through our application to search for campaigns they might be interested in. Once a donor finds a campaign that they like, they can support that campaign by contributing some amount of ether. Then, a Metamask pop up appears confirming the transaction. This donor now becomes a Backer for that campaign and plays a part in deciding whether the vendor is allowed access to the funds raised so far.
4. *Withdrawal Request:* When a vendor wishes to withdraw a certain amount of ether from the pool of money raised so far for their campaign, he must first create a withdrawal request. This request must be approved by at least half of the total backers of that campaign. If the 50% voting criteria is not met, the vendor cannot withdraw the funds and has to wait for the rest of the backers to vote.
5. *Approval:* When a vendor makes a request for spending money, all the backers will be notified about the Withdrawal Request. So, the backer needs to approve the request if he wants. One contributor can give a single vote, i.e., one

approval per backer. A backer can showcase their approval by clicking on the "Vote" button next to the campaign that the vendor has asked the Withdrawal Request for. This will then make a transaction, charging a small gas fee and it will add a block to the blockchain.

6. *Finalize Campaign:* If the vendor can reach the 50\% approval mark, it means that at least half the backers trust and support the campaign and are comfortable with the vendor making the spending request. The ether will be directly transmitted to the verified vendor. In the case where the target contribution amount is not reached, the project will be terminated. All the transactions that have taken place will be stored in the blockchain to provide transparency of the whole process.

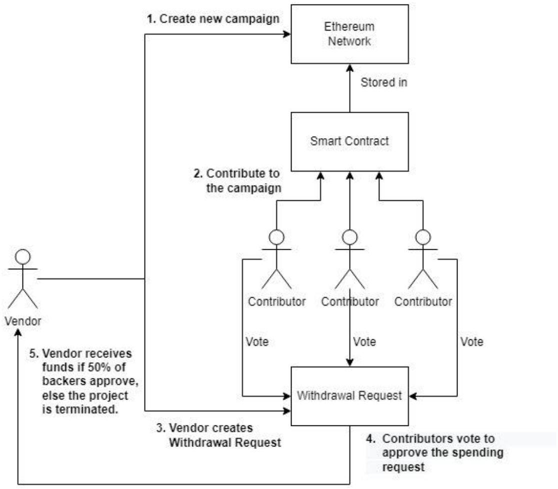

1. Create new campaign
Ethereum Network
Stored in
Smart Contract
2. Contribute to the campaign
Vendor
Contributor
Contributor
Contributor
Vote
Vote
Vote
5. Vendor receives funds if 50% of backers approve, else the project is terminated.
Withdrawal Request
3. Vendor creates Withdrawal Request
4. Contributors vote to approve the spending request

Fig -3: User Flow of the Crowdfunding Application

D. Deployment

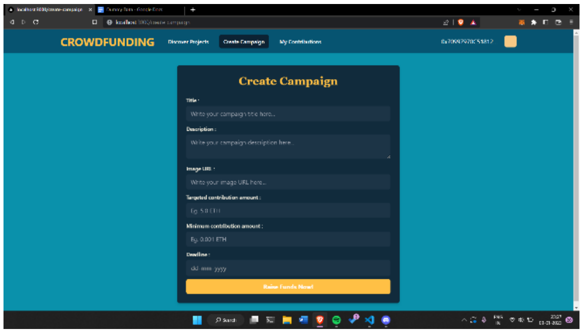
First, we loaded the local blockchain created by Hardhat to test and deploy our smart contract. Our local test node was started with *npx hardhat node*, and this process was continuously run when we deployed our smart contract.

The Crowdfunding instance was then deployed to the Hardhat test network and the address of the deployed Crowdfunding contract was console logged in the terminal using the command *npx hardhat run scripts/deploy.js --network localhost*.

The Crowdfunding application was then started by navigating into the client directory (which contains our frontend code) and running the command *npm run dev*. The application was hosted on localhost:3000.

VIII. RESULTS

After designing the architecture, we made a simple wireframe of our application and implemented it. Then we tested the Web-app, made a few adjustments, and deployed our application. The screenshots of our prototype are given below.



Create Campaign

Title :
Write your campaign title here..

Description :
Write your campaign description here..

Image URL :
Write your image URL here..

Targeted contribution amount :
Eg. 5.0 ETH

Minimum contribution amount :
Eg. 0.001 ETH

Deadline :
dd-mm-yyyy

Raise Funds Now!

Fig -4: Campaign Creation Form

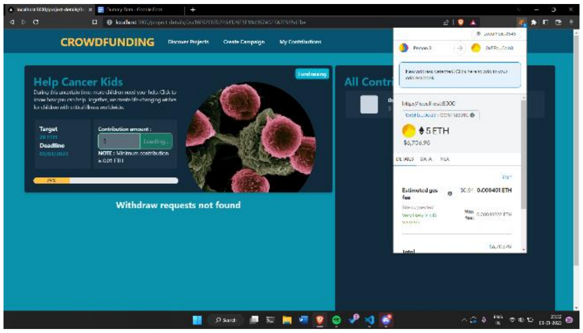


Fig -5: Contributor Contributing 5 ETH to a Campaign

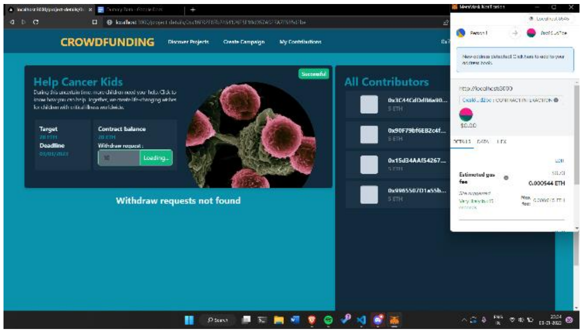
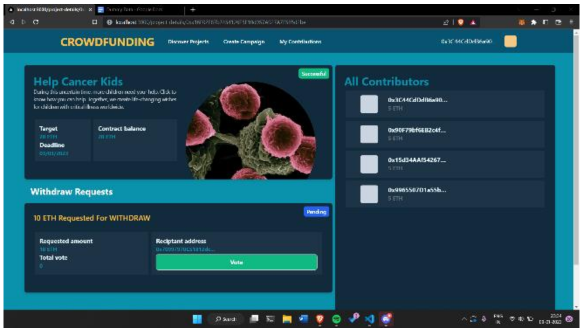


Fig -6: Vendor Creating a Withdrawal Request for 10 ETH



Help Cancer Kids

During this uncertain time, more children need your help. Click to know how you can help. Together, we create life-changing wishes for children with critical illness worldwide.

Target 20 ETH
Deadline 05/01/2023
Contract balance 20 ETH

Withdraw Requests

10 ETH Requested For WITHDRAW

Requested amount 10 ETH
Total vote 0
Reciptant address 0x70997970C51812dc...

All Contributors

0x3C44CdDdB6a90... 5 ETH
0x90F79bf6EB2c4f... 5 ETH
0x15d34AAf54267... 5 ETH
0x9965507D1a55b... 5 ETH

Fig -7: Contributor Giving Single Vote Approval for Campaign Withdrawal Request

CONCLUSION

In this paper, we have realized that integrating blockchain technology with the traditional crowdfunding platform has helped eradicate the many weaknesses present in normal crowdfunding processes by creating a decentralized platform to remove the concept of central authority thereby eliminating the need for intermediaries, providing transparency of transactions, making it affordable for campaign creators to launch their campaign due to low gas fees, improving the security measures to prevent fraudulent actions from taking place and increasing the speed of transactions since each transaction takes only a couple of minutes to complete.

Blockchain technology keeps evolving and advancing as the days go by. With the introduction of initial coin offerings (ICOs) [19], our application has a lot of scope for improvement. Blockchain is still very new in the tech industry and the world is still adapting to it.

ACKNOWLEDGMENT

We would like to thank Mr. Yogesh Sharma for his valuable comments and suggestions to improve the quality of the paper. We are also grateful to Mr. Sandeep Tayal for helping us review our work regularly. We would also like to thank the Department of Computer Science and Engineering, Maharaja Agrasen Institute of Technology for providing us with the knowledge and infrastructure for us to build and test our code.

REFERENCES

[1] Lee, Jei Young. "A decentralized token economy: How blockchain and cryptocurrency can revolutionize business." Business Horizons 62.6 (2019): 773-784.

[2] Kshetri, Nir. "Will blockchain emerge as a tool to break the poverty chain in the Global South?." Third World Quarterly 38.8 (2017): 1710-1732.

[3] Sarmah, Simanta Shekhar. "Understanding blockchain technology." Computer Science and Engineering 8.2 (2018): 23-29.

[4] Al-Saqaf, Walid, and Nicolas Seidler. "Blockchain technology for social impact: opportunities and challenges ahead." Journal of Cyber Policy 2.3 (2017): 338-354.

[5] Mollick, Ethan. "The dynamics of crowdfunding: An exploratory study." Journal of business venturing 29.1 (2014): 1-16.

[6] Antonopoulos, Andreas M., and Gavin Wood. Mastering ethereum: building smart contracts and dapps. O'reilly Media, 2018.

[7] Arnold, Laurin, et al. "Blockchain and initial coin offerings: blockchain’s implications for crowdfunding." Business transformation through blockchain. Palgrave Macmillan, Cham, 2019. 233-272.

[8] Nguyen, Loan TQ, et al. "The role of blockchain technology-based social crowdfunding in advancing social value creation." Technological Forecasting and Social Change 170 (2021): 120898.

[9] Sobti, Rajeev, and Ganesan Geetha. "Cryptographic hash functions: a review." International Journal of Computer Science Issues (IJCSI) 9.2 (2012): 461.

[10] Mik, Eliza. "Smart contracts: terminology, technical limitations and real world complexity." Law, Innovation and Technology 9.2 (2017): 269-300.

[11] Lee, Wei-Meng. "Using the metamask chrome extension." Beginning Ethereum Smart Contracts Programming. Apress, Berkeley, CA, 2019. 93-126.

[12] Konshin, Kirill. Next. js Quick Start Guide: Server-side rendering done right. Packt Publishing Ltd, 2018.

[13] Panda, Sandeep Kumar, and Suresh Chandra Satapathy. "An investigation into smart contract deployment on ethereum platform using Web3. js and solidity using blockchain." Data Engineering and Intelligent Computing. Springer, Singapore, 2021. 549-561.

[14] Modi, Ritesh. Solidity Programming Essentials: A beginner's guide to build smart contracts for Ethereum and blockchain. Packt Publishing Ltd, 2018.

[15] Palechor, Luisa, and Cor-Paul Bezemer. "How are Solidity smart contracts tested in open source projects? An exploratory study." (2022).

[16] Yuan, Qi, et al. "Detecting phishing scams on ethereum based on transaction records." 2020 IEEE International Symposium on Circuits and Systems (ISCAS). IEEE, 2020.

[17] Wathan, A. "Tailwind CSS: A utility-first CSS framework for rapid UI development." (2021).

[18] Banks, Alex, and Eve Porcello. Learning React: functional web development with React and Redux. " O'Reilly Media, Inc.", 2017.

[19] Moxoto, Ana Claudia De, Paulo Melo, and Elias Soukiazes. "Initial Coin Offering (ICO): a systematic review of the literature." Proceedings of the 54th Hawaii International Conference on System Sciences. 2021.

[20] D’Ambrosio, Mario, and Gianfranco Gianfrate. "Crowdfunding and venture capital: Substitutes or complements?." The Journal of Private Equity 20.1 (2016): 7-20.

[21] Baici, Eliana, Laura Capraro, and Martin Zagler. "Lemmings in the Crowd: Success and Failure of Crowdfunding Platforms." Lemmings in the Crowd: Success and Failure of Crowdfunding Platforms (2017): 337-349.

[22] Amin, Md Ratul, and Megat F. Zuhairi. "CROWDFUNDING SMART CONTRACT: SECURITY AND CHALLENGES."

[23] Banafa, Ahmed. "31 How Blockchain is Revolutionizing Crowdfunding." (2022): 163-166.

[24] Olivier, Starkenmann, Karl Schmedders, and José Parra Moyano. "Implementation of a Crowdfunding Decentralized Application on Ethereum Master Thesis."

[25] Baber, Hasnan. "Blockchain-based crowdfunding." Blockchain Technology for Industry 4.0. Springer, Singapore, 2020. 117-130.