1. INTRODUCTION

1.1. Preamble

Crowdfunding is the process of collecting contributions from a large number of people for creative projects on websites like Kickstarter and Indiegogo. It typically lasts just a few months, or even less frequently, a few weeks.

Three different types of on-screen characters can be found on the current crowdfunding platform: the Idea Person, who launches a campaign by sharing their creative ideas, the Contributor, who donates money to a specific idea, and the Central Authorized Platform, which acts as an interface for both the Idea Person and Contributor to ensure a project's success.

The current crowdfunding systems have a number of issues. The platform is controlled by a single entity, so there is a significant chance that data will be sold or leaked. To process the requests, traditional crowdfunding sites charge a significant fee to both the Idea person and the Contributors. Many fraudulent actions that cost contributors money have been documented in recent years.

Blockchain-based crowdfunding aids in overcoming the drawbacks of traditional crowdfunding. A digital ledger that is decentralized and distributed called a blockchain is made up of blocks that are transparent and immutable. These blocks include a certain number of transactions and are hashed using cryptographic techniques. Blockchain crowdfunding differs from conventional crowdfunding in that it uses decentralized nodes to mine the blocks with the aid of a consensus mechanism.

With the use of blockchain technology, users may send transactions and create applications on a decentralized network without the need for a server or a centralized authority.

It helps to eliminate central authority and provides security, and transparency. The proposed system introduces a blockchain-based crowdfunding platform that provides a valid Idea Person to create a campaign and the interested users contribute to the campaign, it helps to build trust between the person with the idea and the contributors.

1.2. Motivation

Many centralized crowdfunding systems are existing, among them Kickstarter is also one in which 8% of the contributed amount went to failed projects. There is a need to eliminate central authorities like admin in this scenario as there is a high risk of data getting leaked or sold. Blockchain's primary unique selling point is its decentralized structure. By lowering the processing costs, it may have an impact on crowdfunding campaigns. Blockchain can significantly lower the cost of crowdfunding for creators since it eliminates the need for middlemen or other third parties in financial transactions. Hence helps to build trust and transparency.

1.3. Objectives

1. To build an interactive website for the users.
2. To develop an algorithm to create smart contracts for the crowdfunding project.
3. To construct a voting process, so that only contributors can either approve/reject the spending request

1.4. Literature Survey

[1] Uses a new framework to insure a cyber product using blockchain smart contracts. The vendor can initiate a request for insuring a cyber product, then the interested insurers participate in a sealed-bid auction by bidding their preferred premium for insurance service. Auction winners are selected as insurers and in return, tokens are given. The Auditor will be checking and validating the crowdfunding campaign for any fraudulent activities, but how the auditor will validate the crowdfunding campaign manually is not mentioned and as it is a manual process to validate the campaign there is a chance of fraud activities that could occur hence lack of security and safety.

In the Likestarter [2] a Decentralized Autonomous Organization (DAO) is structured which is based on Ethereum. The system allows registered users to share songs, videos, or other artifacts. Donors when a certain amount of ether is converted into tokens, in return the donor will also gain a token which is introduced by this platform ie: Likoin. The likoin provides the donor the power for voting and it can be converted into bucks to buy an artifact. And the Likoins can be shared among the users.

B.M.A.L Basnayake [3] purposed an idea for verifying food quality and tracking the agricultural supply chain by implementing a blockchain-based solution. A concept of public blockchain was introduced instead of private blockchain in this paperwork any person can access the network ensuring transparency. For each physical product, Ethereum smart contracts were developed and deployed. A unique quick response code was incorporated to identify each and every product in the supply chain uniquely. To deploy the product contract only the farmers are eligible.

In the crowdfunding application proposed by [4] which uses Ethereum smart contracts, a campaign or project idea can be created by the project manager. The project manager will also specify a minimum amount that should be contributed to that particular project. The user has to contribute the minimum amount to become a Contributor and to get voting power. When the project manager sends a spending request with money to be spent and the vendor's wallet address, the contributors will either accept or decline the request by voting. If more than 50% of contributors agree and accept the request then only the request can be finalized by the Project manager. This proposal reduces fraud activities but it has some limitations, such as no time limit given for the contributor to either accept or reject the request for which the project manager may have to wait until a majority is achieved.

The crowdfunding platform [5] introduced by Sujaritha M and others motive to provide a safe and trustable platform for the platform users. It builds a trust that the money or fund contributed can not be stolen or misused in a wrong way. They have developed the application using smart contracts to handle transactions and metadata safely. It ensures the investor's control over how their money or fund has been used. But only one observation is that if users lose their wallet's private key or mnemonic phrase, no way to retrieve the funds inside it.

1.5. Problem Definition

To design and develop a decentralized crowdfunding website using smart contracts.

Description:

Crowdfunding is the process of collecting contributions from a large number of people for creative projects on websites like Kickstarter and Indiegogo. It typically lasts just a few months, or even less frequently, a few weeks.

The current crowdfunding systems have a number of issues. The platform is controlled by a single entity, so there is a significant chance that data will be sold or leaked. To process the requests, traditional crowdfunding sites charge a significant fee to both the Idea person and the Contributors. Many fraudulent actions that cost contributors money have been documented in recent years. So by using Blockchain Technology the issues with existing systems can be overcome.

2. PROPOSED SYSTEM

2.1. Description of Proposed System

In the proposed system there are mainly two actors on the scene who will be interacting with the system and performing tasks, the actors are

- Contributors
- Idea Person (person with the idea)

Both contributors and project initiators must have an Ethereum wallet to perform tasks. When the project initiator creates a campaign for his project a contract will be deployed for his campaign where contributors can fund the project through crypto.

When project initiators create a spending request for their funded project the request cannot be granted, instead it will be verified by the contributors as shown in the diagram below. Based on the requests which are verified and accepted by the contributors the request will be processed if the request gets the majority of votes the amount will be sanctioned for the vendors who provide services for the project, if the request doesn't get sufficient votes then the spending request will be rejected.

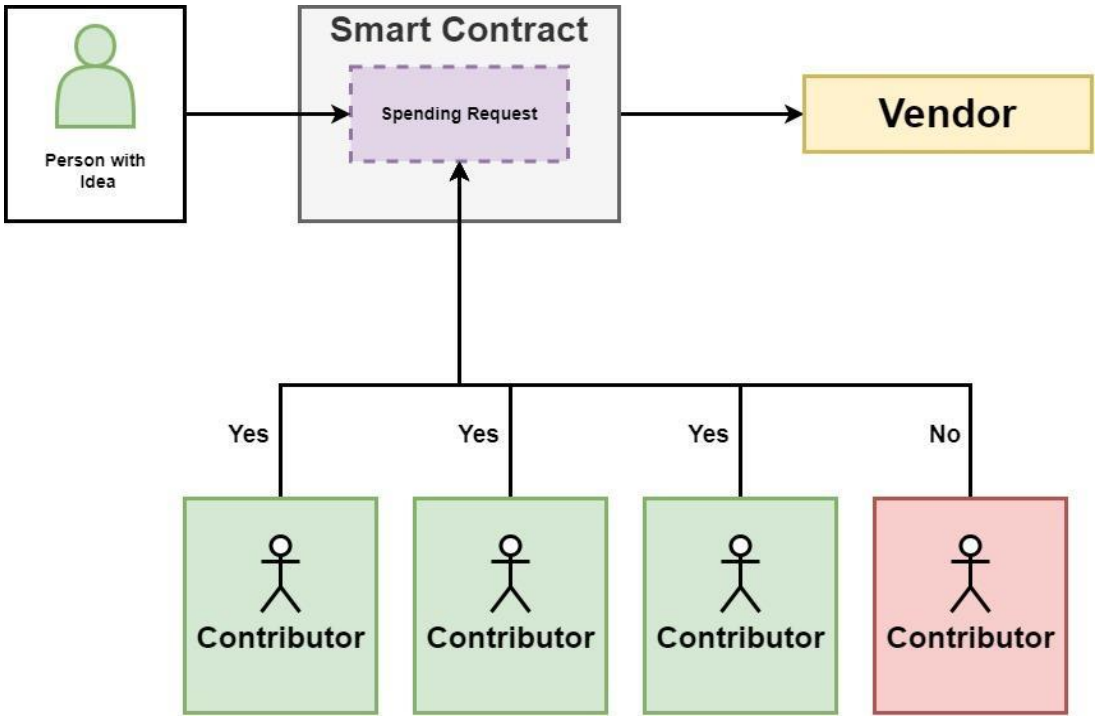

Figure 1. Block Diagram

2.2. Description of Target users

Idea person: The person with the idea creates a campaign for funding. When the person wants to spend the funds to acquire any requirement related to the project he sends a request to the contributors, when more than 50% of the contributors agree to the request then the Idea person can finalize or cancel the payment to the vendors.

Contributor: The person who is interested in funding a project are the contributors. When the Idea person sends a request to spend funds for any requirement then the contributors have the power to approve or reject the request.

2.3. Advantages of Proposed System

Currently we see many fraud activities happening in the existing centralized system, where a central authority will be handling the transactions. In the proposed system we are aiming to build a platform with a decentralized system using blockchain smart contracts.

The process after creating a campaign is not known and once the funds are contributed the contributor does not have control over the system so in the new system the contributor can approve or reject the spending request of the funds sent by the Idea person. By doing so we can develop trust between the contributor and the Idea person.

To minimize fraud activities, In the proposed system when the Idea person wants to spend the funds to acquire any requirement related to the project he sends a request to the contributors, when more than 50% of the contributors agree to the request then only the idea person can finalize or cancel the payment to the vendors.

2.4. Scope of Blockchain-based Crowdfunding Platform

- The transactions are of cryptocurrency only.
- System will be deployed on Ethereum testnet.

3. SOFTWARE REQUIREMENT SPECIFICATION

3.1. Overview of SRS

A system or software application's characteristics and behaviour are described in a set of documents called a system requirements specification (SRS). It consists of a number of components that make an effort to specify the intended functionality needed by the client and to satisfy the various users. The goal is to compile and evaluate all of the many concepts that have been proposed to define the system and its consumer-related requirements. It must completely describe the detected application's or subsystem's external behavior. In order to provide a thorough and in-depth description of the needs for the software, it also discusses nonfunctional requirements, design restrictions, and other elements.

Users and software designers can communicate using the SRS. The SRS's particular objectives are

- Assisting with reviews
- Defining the work's scope
- Giving software developers a reference (i.e. navigation aids, document structure)
- Constructing a testable framework for primary and secondary use cases
- Features that take into account user needs
- Establishing a framework for continuous improvement (via incomplete specs or questions)

3.2. Requirement Specifications

3.2.1. Functional Requirements

- The System shall be able to provide:
    - User to create a campaign.
    - User to create a request for funds.
    - User to finalize the request
    - User to send the requested amount to the vendors.
    - List of campaigns to contributors
    - Contributors contribute to the user’s projects.
    - Contributors to approve the request.

3.2.2. **Use case Diagrams**

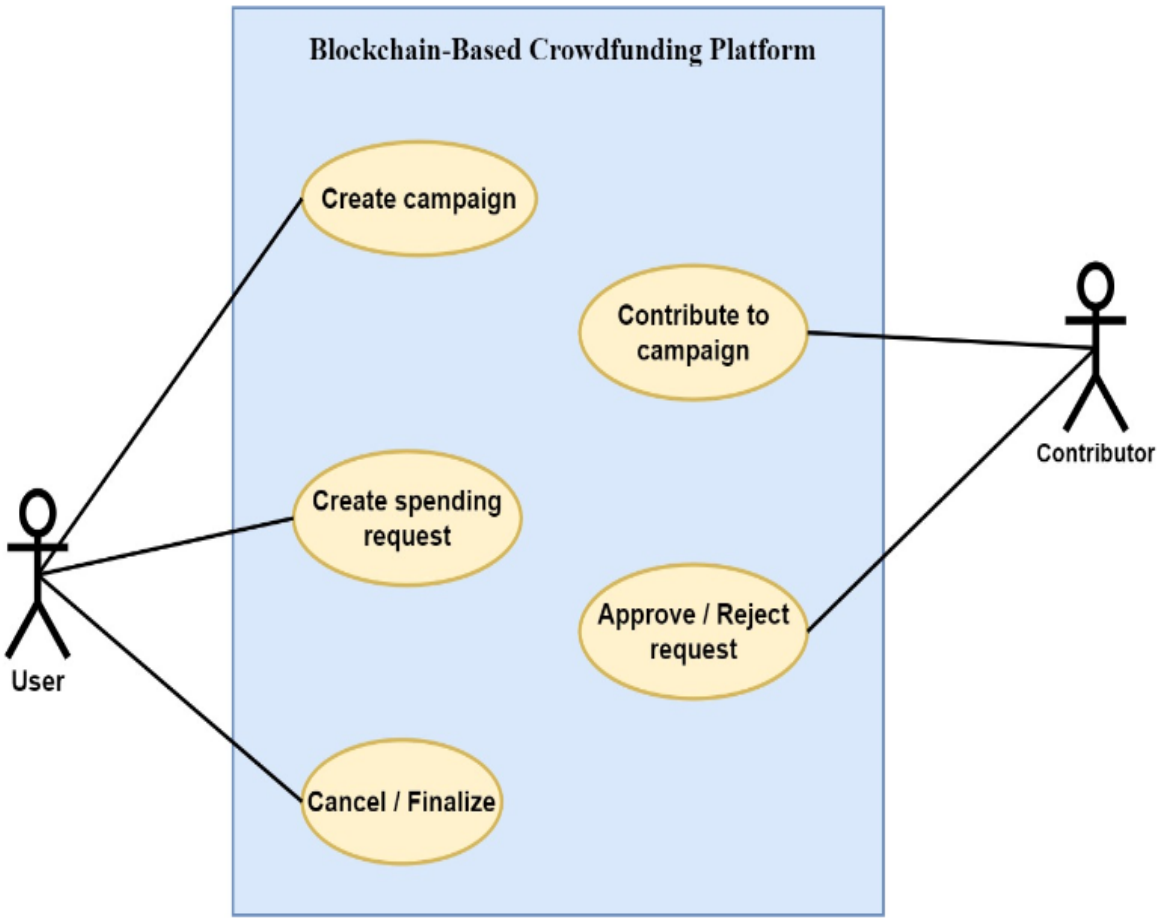

Figure 2. Use Case Diagram

Use case diagram of the system is shown in Figure 2. The two main actors are the Idea person and the Contributor. The Idea person shall be able to create campaigns, and spending requests and can finalize or cancel the request. The contributor can contribute to the campaign and have the power to approve or reject the request.

3.2.3. **Use case descriptions using scenarios**

Use Case: The System shall be able to provide users to create a campaign.
Actors: Project initiator
Pre condition: Must have Ethereum wallet
Post condition: Campaign will be created
Main Success Scenario: 1. User enters the campaign details 2. Sets minimum contribution to his campaign 3. Enter create campaign
Exception Scenario: 3. Displays a message if the balance in the wallet is insufficient

Use Case: The System shall be able to provide user to send the spending request
Actors: Project initiator
Pre condition: 1. Must have Ethereum wallet 2. A campaign must be created 3. Contributors must be contributed to the particular campaign
Post condition: Spending request will be sent to all the contributors
Main Success Scenario: 1. User enters vendor's details 2. The user specifies the amount to be spent
Exception Scenario: 1. Displays message if vendor’s details are not filled 2. Display the message if the amount is more than the amount contributed

Use Case: The System shall be able to provide users to finalize the request
Actors: Project initiator
Pre condition: 1. Must have Ethereum wallet 2. Contributor must have approved the request
Post condition: The amount will be sent to the vendor
Main Success Scenario: 1. Click on Finalize Button
Exception Scenario: No Exception

Use Case: The System shall be able to provide contributor to contribute to the user's campaign
Actors: Contributor
Pre condition: Must have Ethereum wallet
Post condition: Amount will be transferred to the smart contract
Main Success Scenario: 1. Search for the campaign 2. Click on the campaign in which the contributor is interested 3. Enter the amount to be contributed 4. Click on contribute
Exception Scenario: 3.a. Displays a message if the entered amount is less than the minimum amount specified b. Display a message if the amount entered is exceeding the balance in the wallet

3.2.4. Non-functional Requirements

The proposed system should be able to create a campaign within 15 seconds of the user's request.

3.3. Software and Hardware Requirements

- Front end - To fulfill the needs according to the user, the user interacts with the front end of any website. The user can access the functions of the blockchain technology in this way and take part in transactions.
    - HTML5
    - CSS
    - Node js
    - React.js
    - Next.js
- Back end-
    - Solidity programming
    - Web3.js
- Ethereum blockchain:- Our application platform runs on a powerful framework ie: Ethereum blockchain. Smart contracts can be deployed on this framework. These smart contracts contain functions that can be called for various purposes through the front end.
- Ganache CLI:- It is used to connect our front-end application to the Ethereum network.
- Web3 and Metamask:- To interact with Ethereum nodes we have used Web3. For performing transactions on any blockchain network, a person needs to have a wallet. And this need is handled by a meta mask. Through this service, all types of payments can be carried out successfully.

4. SYSTEM SPECIFICATION

4.1. Architecture of the system

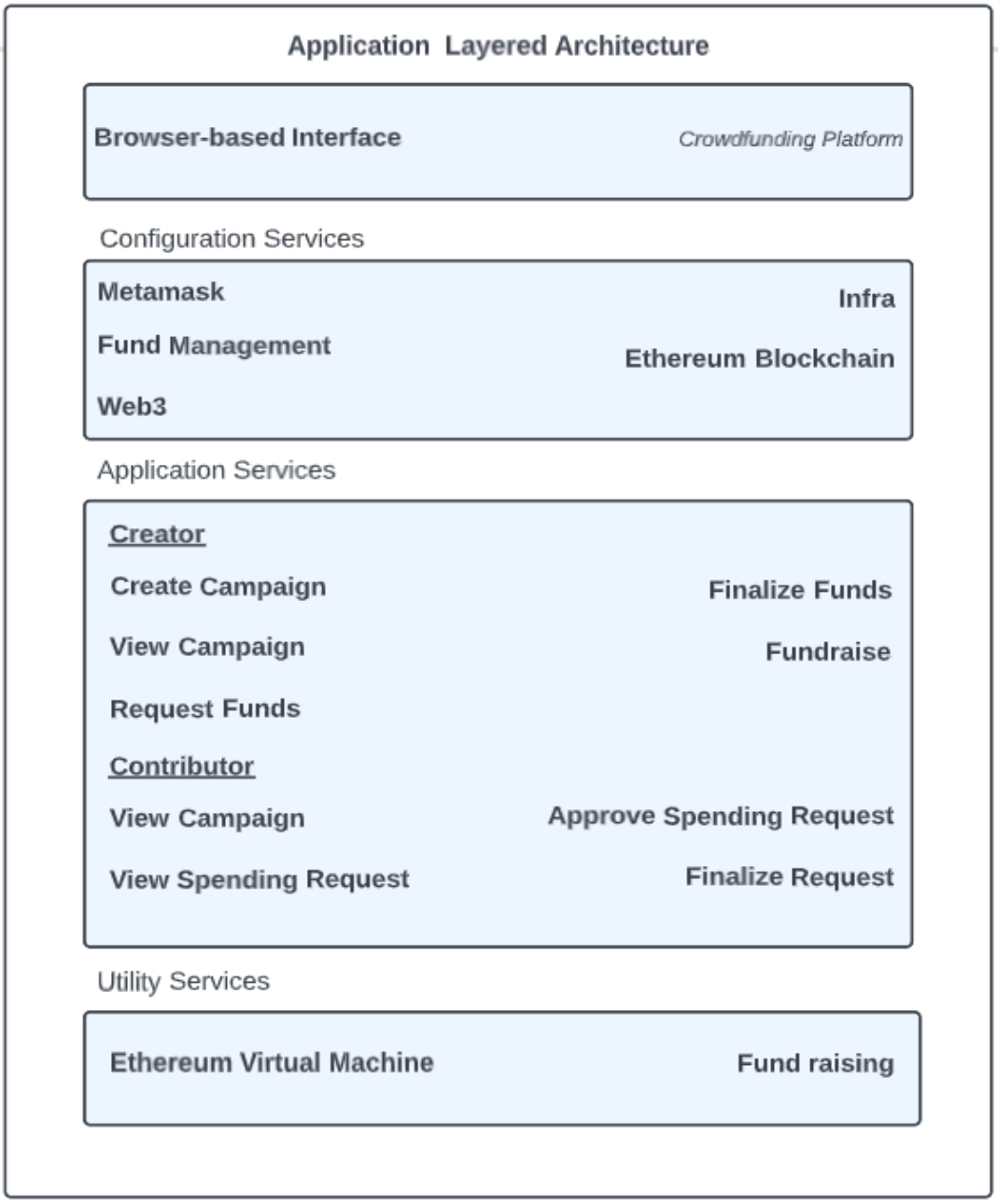

Figure 3. Layered Architecture

- In our project we have used Application Layered Architecture.
- Our project is a browser-based interface called Crowdfunding Platform.
- The configuration services
    - Fund Management: smart contracts are used to manage funds by keeping track of every transaction related to the campaign
    - Web3 Provider Metamask
    - Using ethereum test network for deployment.
- The application services provide different services based on different users.
    - If the user is a creator: who wants to create a campaign, he can be able to create it, by providing campaign details. The creator can also request funds for his campaign. He can also finalize the request if more than 50% of contributors approve the request.
    - If the user is a contributor: who can contribute to the campaign, gains the power of voting and approving the spending request only if the contributor contributed an amount more than the minimum contribution. The contributor can approve or reject the spending request.
- The utility services
    - Ethereum Virtual Machine: When a new block is added to the chain, a software called the Ethereum Virtual Machine, or EVM, computes the state of the Ethereum network and executes smart contracts.
    - Fund raising: A procedure of collecting funds or money in small amount from different people for a project idea or a cause.

5. IMPLEMENTATION

5.1. Proposed Methodology

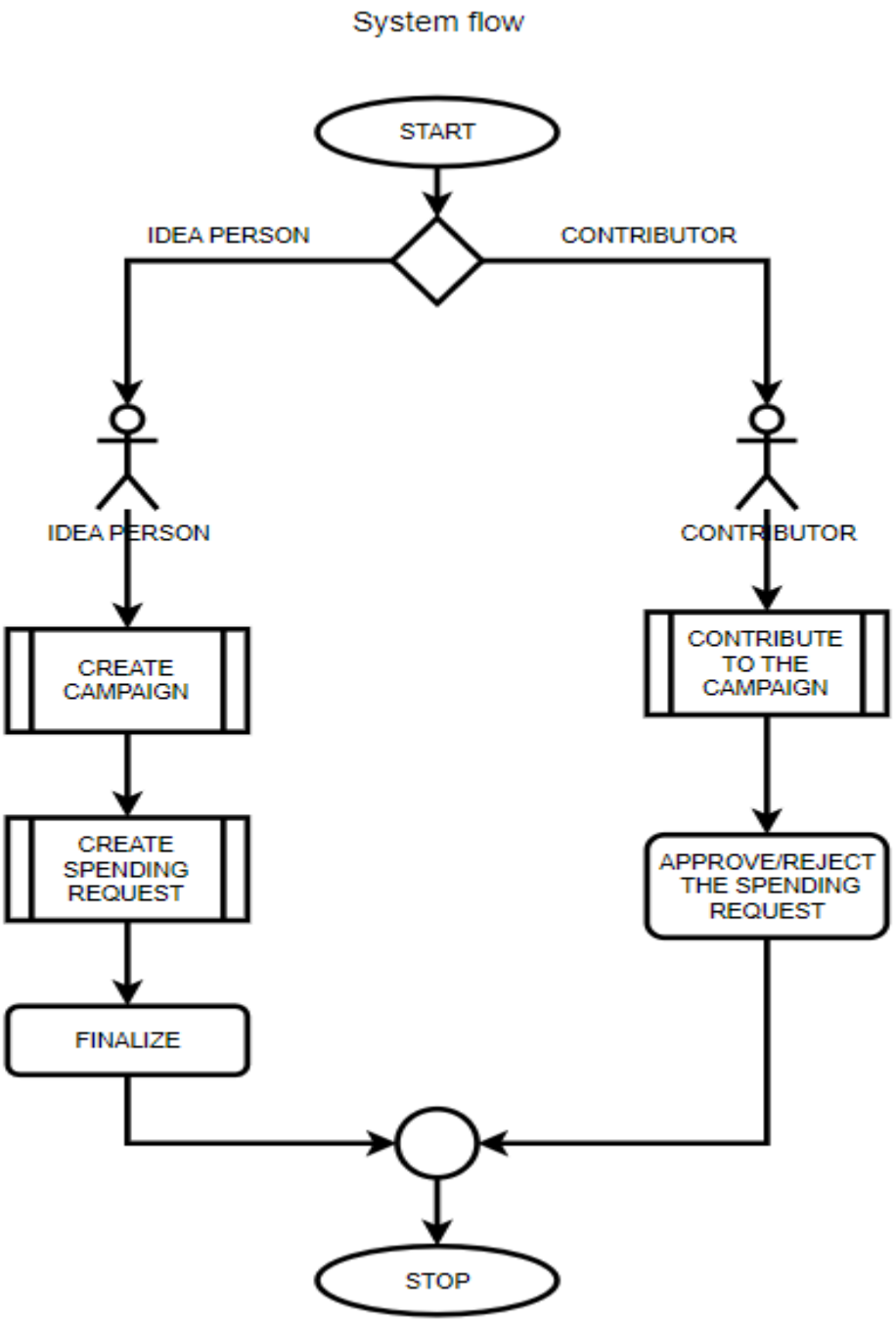

Figure 4. System flow

- An Idea person can create a campaign by proposing the cause or the idea and minimum contribution should be mentioned.
- Once the campaign is created by the Idea person, it is visible to other users. Users who are interested to contribute to a particular project can browse the details of the project or campaign.
- The user can contribute a minimum contribution to the campaign to gain the role of the contributor and the fund will be in the smart contract wallet. It ensures that the Idea person can not misuse the funds, hence the chance of fraudulent activities is not there.
- The Idea person can create a spending request which will be sent to the contributors of the campaign.
- A voting process is conducted where the contributors either approve or reject the spending request sent by the idea person, if more than 50% of the contributors agree and approve the request then only the idea person can finalize or cancel the request.
- When the request is finalized the fund is transferred directly to the wallet of the vendor.

5.2. Module description

5.2.1. Idea Person

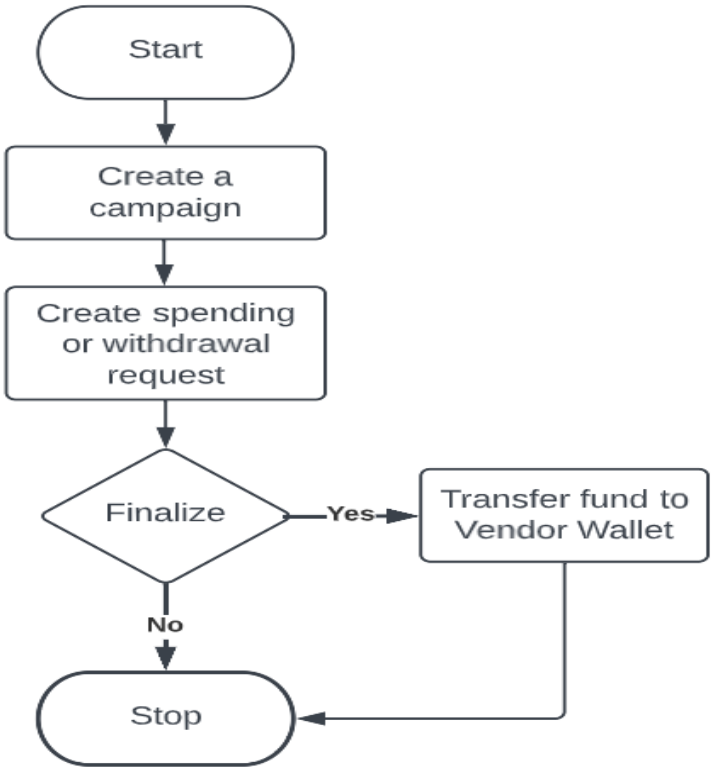

Figure 5. Idea Person

- The user can create a new campaign by adding the details.
- The user should set the minimum contribution needed for the campaign and also should mention the target fund of the campaign.
- Once the target is fulfilled, the Idea person can create a request to withdraw or spend the fund for a requirement.
- Idea person should provide the details of the vendor ie: wallet address of the vendor.
- If maximum contributors approve the request, then the Idea person can finalize the request and the funds specified will be directly transferred to the vendor's wallet.

5.2.2. Contributor

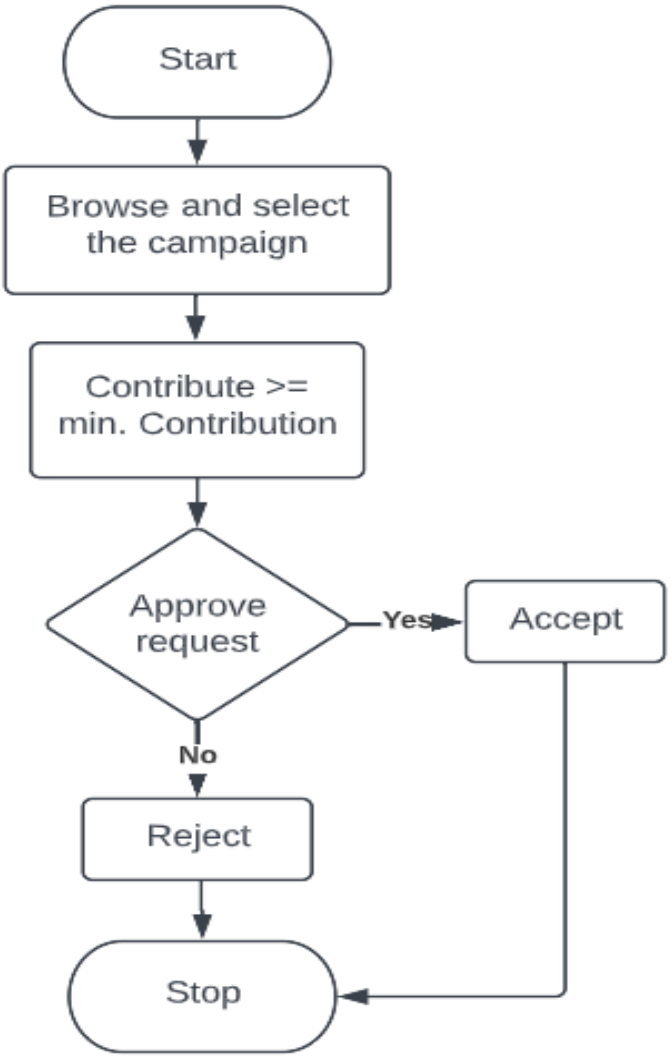

Figure 6. Contributor

- The user browses the website and can check the details of the campaign in which they are interested.
- The user to become the contributor should contribute a minimum contribution as set by the Idea person of the project.
- By becoming the contributor, the user gains the power of voting for the spending request sent by the Idea person.
- If the contributor approves the request then a positive vote is considered.

6. TESTING

6.1. Acceptance Testing

Table 1: Acceptance test plan

Test id	Input Description	Expected output	Actual output
T001	Idea Person should be able to create campaign	Campaign Created successfully	Campaign Created successfully
T002	Random user trying to finalize the request	User should not be able to finalize request if the user is not the creator of the campaign	User should not be able to finalize request if the user is not the creator of the campaign
T003	User should be automatically able to connect to metamask wallet	Popup notification metamask for confirmation	Metamask not connected

6.2. Unit Testing

Test id	Input Description	Expected output	Actual output
T001	Contributor attempting to finalize spending request	Contributor should not be able to finalize spending request	Finalized
T002	Idea attempting to create spending request with zero balance	Request Created	Request Created
T003	Contributor should be able to approve request only if he is a approver	Can be able to Approve request	Can be able to Approve request
T004	The Idea Person should be able to finalize a request only if the majority of approvers agree.	Able to finalize only if majority is attend	Able to finalize only if majority is attend

7. RESULTS AND DISCUSSIONS

This project ensures the elimination of major problems in crowdfunding like.

- Third party involvement in crowdfunding platforms.
- Misuse of invested money on campaigns.
- Wastage of the money from campaigns that are abandoned.
- Lack of trust between campaign managers and contributors.

This crowdfunding web application ensures secure transactions without any involvement of a third party with a friendly UI. For each transaction, there is a record with transaction details. The application allows managers to create campaigns without any charges and contributors can fund these campaigns once the campaign is created the manager can create a spending request for the supplies or utilities needed for their campaigns. The contributors verify these spending requests and decide whether to approve or reject these requests for the transfer of money from smart contracts to vendors(providers).

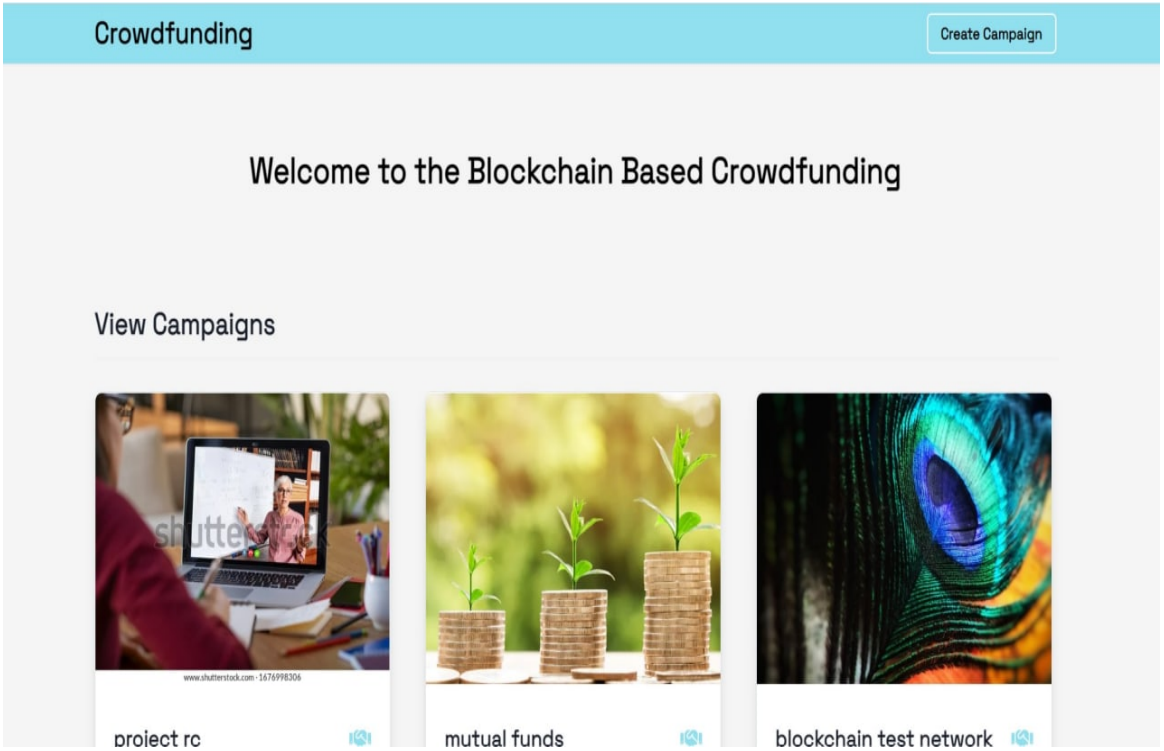

Figure 7.

Welcome page of the website showing different active campaigns and also providing option to create a new campaign for the user.

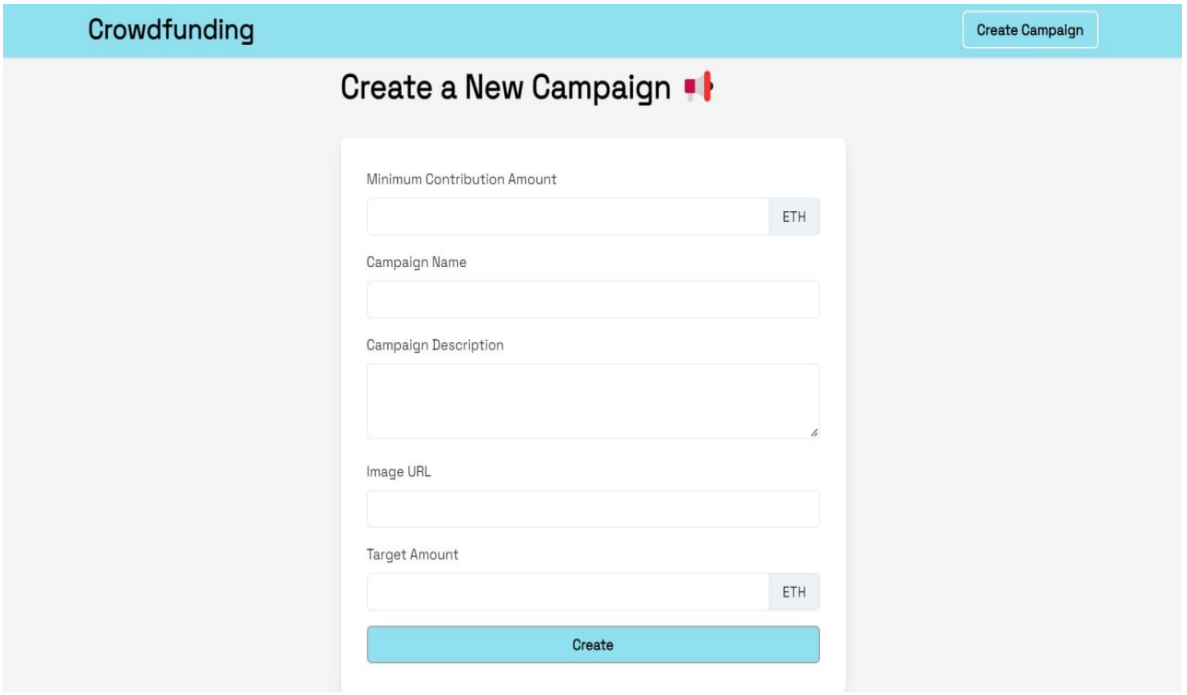

Figure 8.

A form page opens when the user wants to create a new campaign.

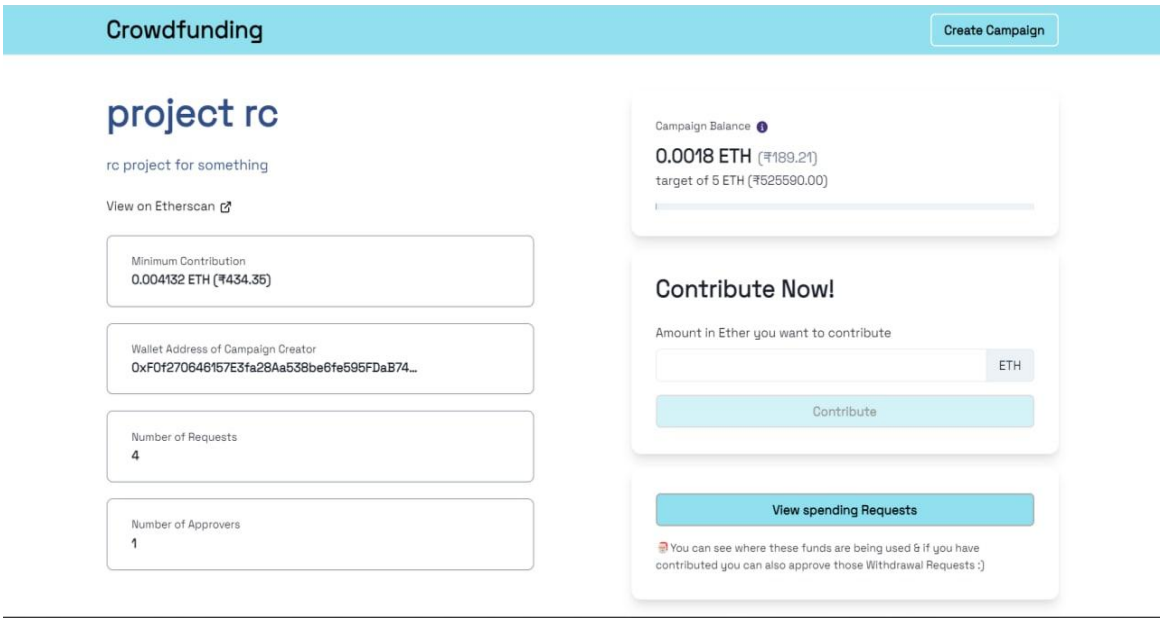

Figure 9.

When a user browses the website and wants to contribute to the campaign. The campaign details will be displayed for the user.

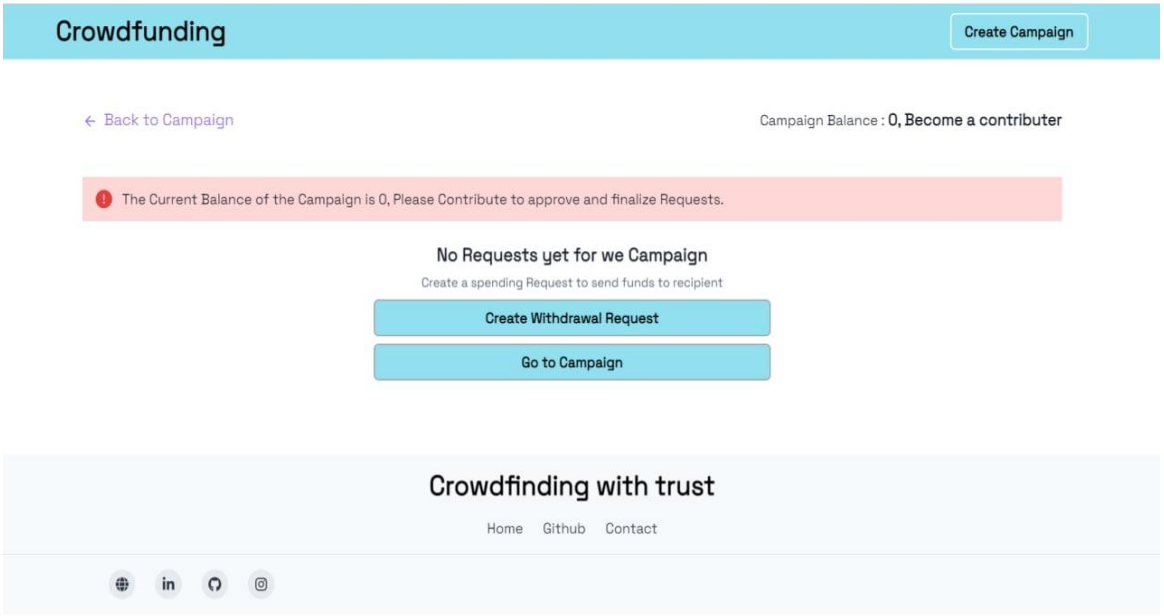

Figure 10.

Idea person can create a request to withdraw or spend for any requirement of the campaign

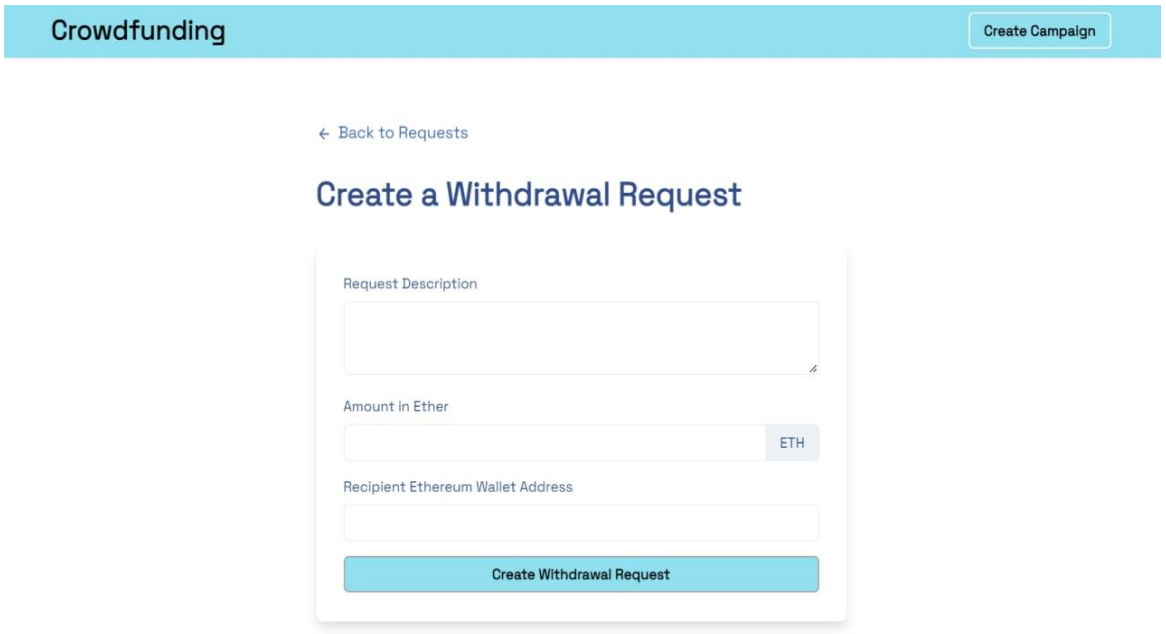

Figure 11.

A form opens for withdrawal request for the Idea person, amount in ethers should also be mentioned by the Idea person.

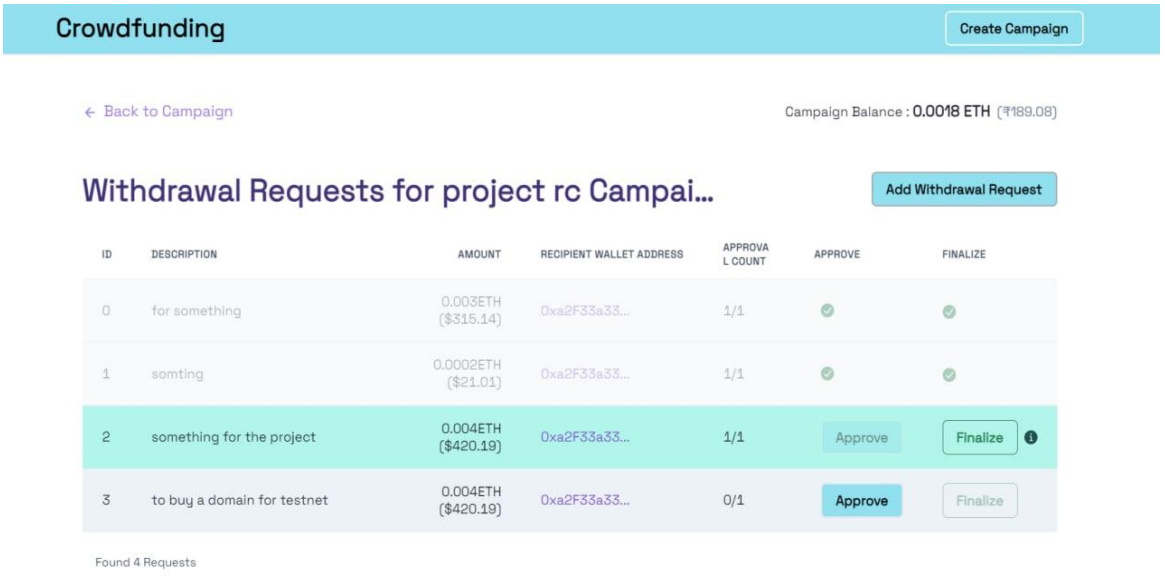

Figure 12.

Once the request is approved by the majority of the contributors, the Idea person can finalize the transaction.

8. CONCLUSION AND FUTURE SCOPE

To conclude, a crowdfunding platform using blockchain technology aims to build trust and transparency between the Idea Person(Manager) and the Contributor, by eliminating the central authority and incorporating smart contracts. The contributor will have control over their money or fund hence they can track their funds and need not worry about the occurrence of any fraud activities.

The proposed idea has a very bright future, as it aims to build trust and transparency. Transactions are of cryptocurrency only. In the future donating through various different cryptocurrencies can be incorporated by implementing functionalities.

9. REFERENCES / BIBLIOGRAPHY

[1] “Crowdfunding the Insurance of a Cyber-Product using Blockchain " Vakilinia, S.Badsha, S.Senugupta. 9th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON) 2018, New York, USA.

[2] “LikeStarter: a Smart-contract based Social DAO for Crowdfunding” M. Zichichi, M. Contu, S. Ferretti and G D’Angelo. 2019 IEEE Conference on computer communications workshops (INFOCOM WKSHPS), Paris, France.

[3] “A Blockchain-based decentralized system to ensure the transparency of the organic food supply chain” B. M. A L. Basnayake, C. Rajapakse. Research conference of smart computing and systems engineering (SCSE), IEEE, 2019, Colombo, Sri Lanka.

[4] “Venturing Crowdfunding using Smart Contracts in Blockchain” N. Yadav and Sarasvathi .V. Third International Conference on Smart Systems and Inventive Technology (ICSSIT 2020), Tirunelveli, India.

[5] “Decentralized Crowdfunding Platform using Smart Contracts". Sujaritha M, Shumugapriya S, Arun B S, Hariharan S, Inderajith K. IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems (SPICES) 2022, Coimbatore, India.

[6]“https://remix.ethereum.org/#optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.7+commit.e28d00a7.js”

10. APPENDIX

A. Gantt Chart

B. Glossary

Blockchain:
Blockchain is a decentralized system, which has distributed public digital ledger that will be used to record various transactions across many computers. It is very difficult or impossible to alter or hack the record.

Smart Contracts:
A Smart Contract is a program or a transaction protocol that will automatically execute, and controls the events that should be carried out. The event will be defined in form of an agreement.

Ganache CLI:
It is used to connect our front-end application to the Ethereum network.

C. **Description of Tools & Technology used**

Blockchain Technology:

With the use of blockchain technology, users may send transactions and create applications on a decentralized network without the need for a server or a centralized authority.

Tools

Remix IDE: Remix IDE is a no-setup smart contract development tool with a GUI. Remix will get you rolling in double time and is used by both pros and novices. Remix functions well in conjunction with other tools and enables a quick deployment to the chain of your choice. Our visual debugger at Remix is well-known.

Ganache CLI: It is used to connect our front-end application to the Ethereum network.

Truffle: With the goal of simplifying the work of developers, Truffle is a top-notch programming environment, testing framework, and asset pipeline for blockchains running on the Ethereum Virtual Machine (EVM).

Infura: Consensys developed the Infura set of developer tools to assist programmers in connecting their applications to the Ethereum network and other decentralized platforms.

mnemonic
'7d24d754679d783b6957b9eb75f85ebb58103b5ed7dd3980982676101669bdb3'

URL=https://sepolia.infura.io/v3/5b1f777ef6f54a66b1d158c0c2f3cf85

Web3: The phrase "Web3" has come to represent the idea of a new, improved internet. In essence, Web3 uses blockchains, cryptocurrencies, and NFTs to return ownership and authority to the consumers.

basically, we use web3 to compile and deploy contracts in real time and interact with the frontEnd UI

Team Number	N3
Guide	Mrs. Vidya Handur
Project title	Blockchain-based Crowdfunding Platform
Industry name	-
Department vertical(either of Data/ Network/System)	Network
University/ Department Research group	