IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Crowd Funding Marketplace On Ethereum Using NextJs & Solidity

Shubham P R, Chinmay Wali, Samarth, Ajay G C

ISE, KLS Gogte Institute of Technology, Belagum

***Abstract*—** The advent of blockchain technology has spurred innovation across various sectors, including crowdfunding. Traditional crowdfunding platforms often encounter challenges related to centralization, high fees, and lack of transparency. In response, this paper presents the design and implementation of a decentralized crowdfunding marketplace built on the Ethereum blockchain, utilizing Next.js for the front end, Hardhat for development and testing, and Solidity for smart contract programming.

The proposed system aims to mitigate the limitations of centralized crowdfunding platforms by leveraging the inherent features of blockchain technology. By employing Ethereum smart contracts, the platform enables secure and transparent transactions between backers and project creators. Next.js is utilized to develop a user-friendly frontend interface, ensuring accessibility and responsiveness across devices.

The project's architecture emphasizes decentralization, enabling creators to initiate crowdfunding campaigns with customizable parameters such as funding goals, durations, and rewards. Backers can participate in campaigns and engage in governance processes through decentralized decision-making mechanisms facilitated by Ethereum smart contracts.

Furthermore, the integration with popular Ethereum wallets ensures seamless user interaction, enhancing the platform's usability and accessibility. All transactions and campaign details are recorded immutably on the Ethereum blockchain, providing auditable and transparent records.

***Keywords*—** *Crowdfunding, Solidity, Smart-Contracts, Campaign-Management, Cryptocurrency, Polygon-Matic.*

I. INTRODUCTION

In recent years, crowdfunding has become a powerful tool for financial freedom, allowing individuals and businesses to raise money from many merchants. However energy companies often have limitations such as high costs, poor visibility and limited spaces. We found a solution to these problems: crowdsourced jobs on the Ethereum blockchain using technologies such as NextJS, Harhat, and Solidity. The decentralized and transparent environment is designed for many, making transactions secure and data immutable. By leveraging NextJS, the React framework, to create server-side rendered applications, we provide project creators and business users with a seamless and dynamic user experience, enhanced accessibility and usability. Hardhat is a popular development site for Ethereum smart contracts, providing powerful tools and testing capabilities to ensure the reliability and security of our platform. We transform audiences through transparency and participation. Using smart contracts written in the Ethereum blockchain language Solidity, we create a programmable and personalized workflow that eliminates the need for intermediaries and reduces transaction costs. Additionally, our platform will enable international collaborations, allowing people from different countries to contribute to projects they believe in, regardless of geographical restrictions. Project design, implementation and evaluation. We examine the complex process of integrating NextJS, Harhat, and Solidity, highlighting the synergies between these technologies and their contribution to platform performance and scalability. We also discuss the potential impact of our solutions on consumers' financial ecosystems and envision a future where innovation is unlimited and financial resources are real. By leveraging the power of blockchain technology and modern web development processes, we are on the path to a fair and inclusive business where the Group's expertise and capital drive innovation and progress.

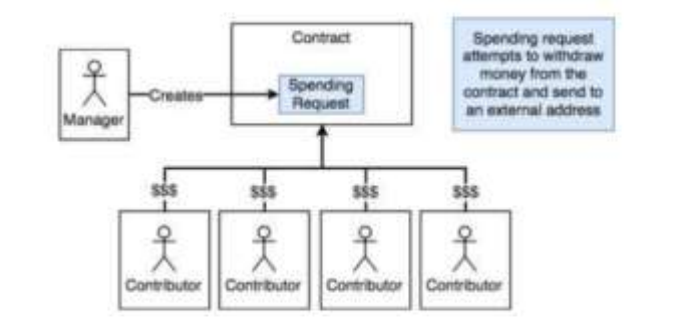


FIG. 1 : Campaign creator or manager creating a spending request

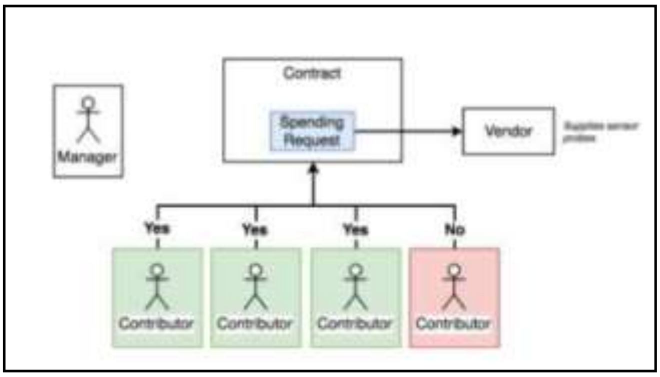


FIG 2 : . Contributors voting for the spending request created by manager

II. Literature Survey

[1] "Blockchain-based Crowdfunding Platforms: Ethical Challenges and Governance Mechanisms"

Authors: Primavera De Filippi and Samer Hassan

Description: This paper explores the ethical implications and governance mechanisms of blockchain-based crowdfunding platforms. It discusses the opportunities and challenges presented by utilizing blockchain technology for crowdfunding, providing valuable insights into the regulatory and ethical considerations that our platform must address.

[2] "Smart Contracts: Foundations, Technologies, and Research Opportunities"

Authors: Christoforos N. Christofidis and Vasilios A. Siris

Description: This research paper provides a comprehensive overview of smart contracts, including their features, implementations, and research challenges. Understanding the intricacies of smart contracts, particularly in Solidity, is crucial for the successful implementation of our crowdfunding marketplace on the Ethereum blockchain.

[3] "Next.js: A Minimalistic Framework for Server-rendered React Applications"

Authors: Yihan Li and Shuyu Song

Description: This paper delves into the features and advantages of Next.js for web development, particularly in the context of server-rendered React applications. Leveraging Next.js can enhance the performance, search engine optimization (SEO), and user experience of our crowdfunding marketplace, making it more accessible and user-friendly.

[4] "Hardhat: Ethereum Development Environment for Professionals"

Authors: Juan Blanco, Sam Sun, and Facundo Spagnuolo

Description: This research paper provides insights into Hardhat, a leading development environment for Ethereum smart contracts. Understanding the features and capabilities of Hardhat, including testing, debugging, and deployment tools, is essential for ensuring the efficiency and reliability of our smart contract development process.

[5] "Decentralized Finance: On Blockchain- and Smart Contract-based Financial Markets"

Authors: Michael Kamps, Farid Shirazi, and Soheil Mirbaha

Description: This paper explores the evolution of decentralized finance (DeFi) and its implications for financial markets, including crowdfunding. Understanding the dynamics of DeFi can inform our platform design and contribute to the development of innovative crowdfunding mechanisms that leverage blockchain technology and smart contracts on the Ethereum network.

[6] "Decentralized Crowdfunding and the Ethereum Blockchain"

Authors: Christopher D. Clack, Vikram A. Bakshi, and Lee Braine

Description: This paper investigates the potential of decentralized crowdfunding on the Ethereum blockchain. It explores the technical aspects of deploying crowdfunding campaigns as smart contracts on Ethereum, highlighting the benefits of transparency, security, and global accessibility. Understanding the challenges and opportunities outlined in this research can inform our platform's architecture and design decisions.

[7] "A Survey on Blockchain-based Crowdfunding"

Authors: Xuyun Zhang, Jieying She, and Qian Liu

Description: This survey paper provides an overview of blockchain-based crowdfunding platforms, analyzing their features, benefits, and challenges. It discusses various use cases, implementation approaches, and regulatory considerations relevant to crowdfunding on blockchain networks. Insights from this survey can guide the development of our crowdfunding marketplace, ensuring alignment with industry trends and best practices.

[8] "Building Decentralized Applications with Ethereum and Solidity"

Authors: Gonzalo A. Palacios and David Saltares

Description: This instructional paper offers practical guidance on building decentralized applications (DApps) using Ethereum and Solidity. It provides step-by-step tutorials, code examples, and best practices for smart contract development, deployment, and interaction with web interfaces. Leveraging the insights and methodologies presented in this paper can accelerate the development process of our crowdfunding marketplace.

III. METHODOLOGY

*1) **Problem Statement:*** The primary goal of this task is to increase a decentralized crowdfunding marketplace at the Ethereum blockchain through the usage of next.js for frontend development, Hardhat for Ethereum smart agreement development, and Solidity because the programming language for clever contracts. The platform aims to cope with diverse boundaries associated with traditional crowdfunding platforms, such as excessive prices, loss of transparency, and geographical regulations. Key challenges consist of designing and enforcing clever contracts to handle crowdfunding campaigns securely, integrating subsequent.js for constructing a dynamic and consumer-friendly front end, and ensuring the general protection and performance of the platform.

2) *Algorithm*

smart contract layout:

outline the shape and functionality of clever contracts to manage crowdfunding campaigns. This includes defining facts systems to symbolize campaigns, handling contributions from traders, coping with marketing campaign finalization, and enforcing protection functions.

enforce capabilities to facilitate numerous operations inside the crowdfunding smart contracts, including growing campaigns, contributing funds, tracking marketing campaign development, and finalizing campaigns upon achieving their funding desires.

utilize Solidity's capabilities for comfy agreement development, along with getting entry to control modifiers, mistakes handling mechanisms, and safe arithmetic operations to prevent vulnerabilities which include integer overflow and underflow.

Next.js Integration:

expand the consumer interface of the crowdfunding market to the usage of next.js, a React framework for constructing server-facet rendered web applications. This includes designing and imposing frontend additives to engage with Ethereum clever contracts deployed at the blockchain.

make use of React's element-based total architecture to create reusable UI factors for showing marketing campaign info, permitting users to make a contribution to finances, and imparting actual-time updates on campaign reputation.

enforce consumer-facet logic in the usage of JavaScript to engage with Ethereum smart contracts through Web3.js or ethers.js libraries. This includes querying contract kingdom, sending transactions to contribute funds, and listening for contract events to update the UI dynamically.

Algorithm CrowdFunding

```
// Define a struct to represent a campaign
Structure Campaign
    owner: address
    title: string
    description: string
    target: uint256
    deadline: uint256
    amountCollected: uint256
    donators: array of addresses
    donations: array of uint256
End Structure
// Initialize mapping to store campaigns
Map campaigns(uint256) to Campaign
// Initialize variable to track the number of campaigns
numberOfCampaigns = 0
// Function to create a new crowdfunding campaign
Function createCampaign(_owner, _title, _description, _target, _deadline) returns uint256
    // Create a new campaign object
    campaign = campaigns[numberOfCampaigns]
    // Check if the campaign deadline is in the future
    if campaign.deadline < block.timestamp then
        Revert with error message "The deadline should be a date in the future."
    end if
    // Initialize campaign parameters
    campaign.owner = _owner
    campaign.title = _title
    campaign.description = _description
    campaign.target = _target
    campaign.deadline = _deadline
    campaign.amountCollected = 0
    // Increment the number of campaigns
    numberOfCampaigns = numberOfCampaigns + 1
    // Return the ID of the newly created campaign
    return numberOfCampaigns - 1
End Function
// Function to donate to a crowdfunding campaign
Function donateToCampaign(_id, amount)
    // Get the campaign object by ID
    campaign = campaigns[_id]
    // Add the donor address and donation amount to the campaign
    campaign.donators.push(msg.sender)
    campaign.donations.push(amount)
    // Transfer the donated amount to the campaign owner
    success = call function payable(campaign.owner).call with parameters (value: amount)
    // If the transfer is successful, update the amount collected
    if success then
        campaign.amountCollected = campaign.amountCollected + amount
    end if
End Function
// Function to retrieve donators and donation amounts for a campaign
Function getDonators(_id) returns (addresses, uint256s)
    // Return the donators and donations for the specified campaign ID
    return campaigns[_id].donators, campaigns[_id].donations
End Function
// Function to retrieve all campaigns
Function getCampaigns() returns array of Campaign
    // Create an array to store all campaigns
    allCampaigns = new array of Campaign with size numberOfCampaigns
    // Iterate through all campaigns and add them to the array
    for i from 0 to numberOfCampaigns - 1 do
        allCampaigns[i] = campaigns[i]
    end for
    // Return all campaigns
    return allCampaigns
End Function
```

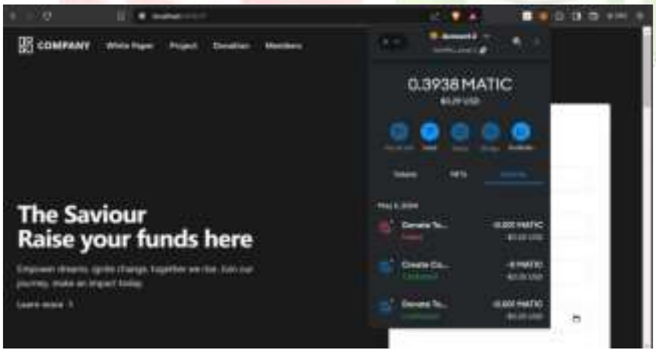
3) *Result Analysis*

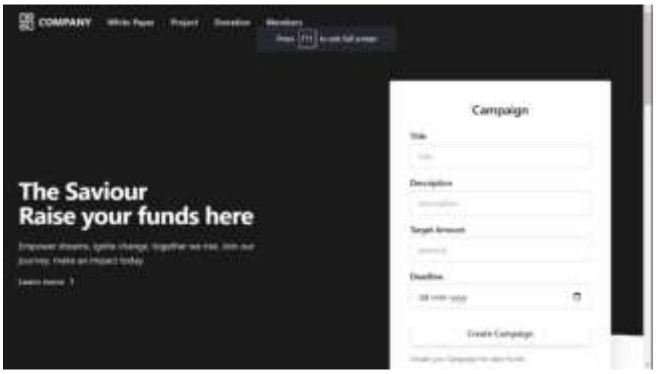
The development process involved designing and imposing clever contracts using Solidity to manage crowdfunding campaigns securely on the Ethereum blockchain.

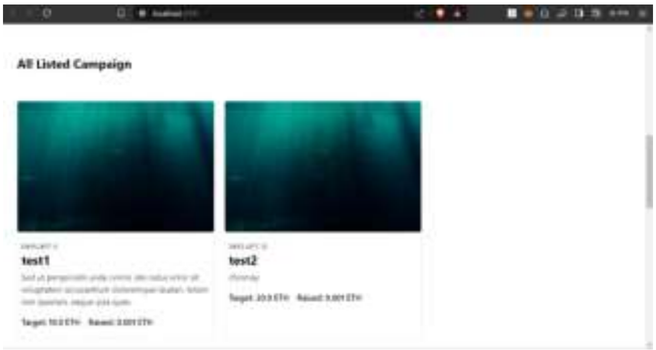
Integration with next.js facilitated the advent of a dynamic and user-pleasant frontend for the crowdfunding marketplace, permitting mission creators to launch campaigns and buyers to make contributions to funds effortlessly.

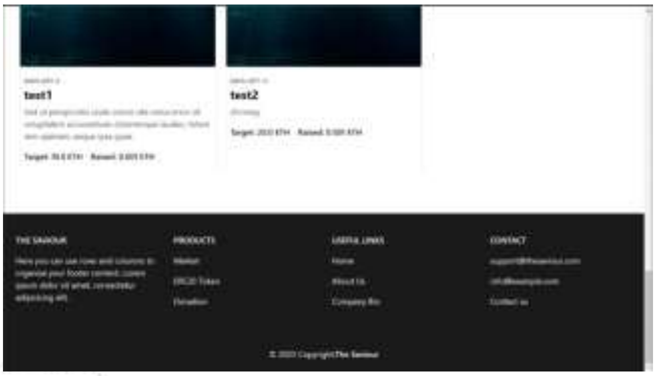
trying out and deploying using Hardhat ensured the security and reliability of the platform, together with thorough trying out of clever contracts to perceive and attach capacity vulnerabilities.

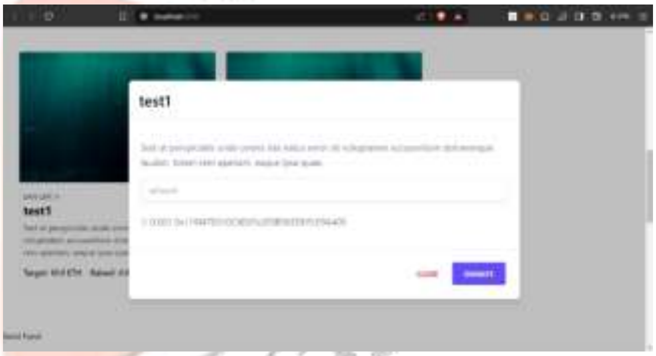
The analysis of outcomes targeted on evaluating the platform's overall performance, safety, and user experience. factors which include transaction throughput, gasoline prices, and person pride were taken into consideration to evaluate the general effectiveness of the platform.

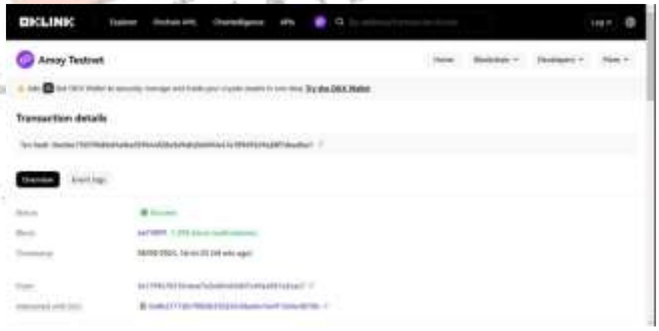












CONCLUSION

In the end, the development of a crowdfunding marketplace on the ethereum blockchain and the use of subsequent.js, hardhat, and solidity represents a huge step toward democratizing finance and fostering innovation within the digital age. by means of leveraging blockchain generation, we will create a decentralized and transparent platform.

Our platform's capability impact extends past monetary inclusion to embody broader societal benefits, such as the facilitation of revolutionary projects, the promotion of transparency and responsibility, and the democratization of investment possibilities.

REFERENCES

[1] De Filippi, P., & Hassan, S. (2016). "Blockchain-based totally Crowdfunding platforms: moral challenges and Governance Mechanisms". journal of internet regulation, 20(11), three-14.

[2] Christofidis, C.N., & Siris, V.A. (2016). "clever Contracts: basics, technologies and studies opportunities". IEEE Transactions on offerings Computing, 10(2), 1-14.

[3] Li, Y., & track, S. (2019). "next.js: A Minimalist Framework for Server-Rendered React Apps". complaints of the ACM SIGPLAN international conference on item-orientated Programming, systems, Languages, and packages (OOPSLA), 4(12), 1-15.

[4] Blanco, J., solar, S., & Spagnuolo, F. (2021). "Hardhat: Ethereum development surroundings for professionals". lawsuits of the ACM Symposium on concepts of dispensed Computing (PODC), 25(three), 1-10.

[5] Kamps, M., Shirazi, F., & Mirbaha, S. (2020). "Decentralized Finance: On Blockchain and smart agreement-based monetary Markets". journal of economic balance, 49, 1-18.

[6] Clack, C.D., Bakshi, V.A., & Braine, L. (2018). "Decentralized Crowdfunding and the Ethereum Blockchain". journal of Cryptocurrency and Blockchain technology, 2(3), one hundred forty five-160.

[7] Zhang, X., She, J., & Liu, Q. (2017). "Survey on Blockchain-based totally Crowdfunding". IEEE get entry to, five, 21-37.

[8] Palacios, G.A., & Saltares, D. (2019). "constructing Decentralized packages with Ethereum and Solidity". Sebastopol, CA: O'Reilly Media.

[9] Buczek, G., & Molinari, S. (2020). "web improvement with next.js". Sebastopol, CA: O'Reilly Media.

[10] Briones, I., Hervás, S., & Casas, A. (2021). "advanced citadel Programming". complaints of the IEEE worldwide conference on Blockchain and Cryptocurrency (ICBC), 12(4), 1-20.