**“Preserving Integrity of fundraising using Blockchain”**

*A project report submitted for the partial fulfillment of academic*

*requirements for the award of Degree*

*in*

**BACHELOR OF ENGINEERING**

**in the Department of**

**INFORMATION SCIENCE & ENGINEERING**

Major Project Phase-2 (IS8C02)

**Submitted by**

|  |  |
| --- | --- |
| Abhishek Shankar | 4NI19IS003 |
| Abhishek Singh | 4NI19IS004 |
| Mohit Kumar | 4NI18IS049 |
| R Naveen Kumar | 4NI18IS071 |

**Under the Supervision of**

Mr. Suhaas K P

Assistant Professor,

Department of I.S.&E.,

NIE, Mysuru

Logo

Description automatically generated

**Department of Information Science & Engineering**

**The National Institute of Engineering**

(An Autonomous Institute under Visvesvaraya Technological University, Belagavi)

Manandavadi Road, Mysuru 570 008, Karnataka, INDIA

**June 2023**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | C:\Users\ADMINI~1\AppData\Local\Temp\nie logo jpg.jpg | **The National Institute of Engineering**  (An Autonomous Institute under Visvesvaraya Technological University, Belagavi)  Manandavadi Road, Mysuru – 570 008, Karnataka, INDIA  • Recognized by AICTE, New Delhi  • Accredited by National Board of Accreditation, New Delhi  • Grant-in-Aid by Government of Karnataka |  | |  |  |

**Department of Information Science & Engineering**

**CERTIFICATE**

Abhishek Shankar 4NI19IS003

Abhishek Singh 4NI19IS004

Mohit Kumar 4NI19IS049

R Naveen Kumar 4NI19IS071

Certified that the project work entitled “DeFi Pay: A futuristic Payment System” carried out by above bonafide students of 8th Semester is submitted in partial fulfillment as part of Major Project phase-2 for the award of Bachelor of Engineering Degree in Information Science and Engineering of The National Institute of Engineering, Mysuru, an autonomous institute under Visvesvaraya Technological University, Belagavi during the academic year 2022-2023. It is certified that all suggestions/ corrections suggested during Internal Assessment have been incorporated in the Report deposited in the departmental library.

The project report has been approved as it satisfies the academic requirements in respect of Project Work prescribed for the award of the said Degree.

|  |  |  |  |
| --- | --- | --- | --- |
| **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Signature of the guide** | **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Signature of HoD** | | **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Signature of the Principal** |
| **Mr. Suhaas K P**  **Assistant Professor, Dept of ISE, NIE, Mysore** | **Dr. Girish**  **Prof. & Head,**  **Dept of ISE, NIE, Mysore** | | **Dr. Rohini Nagapadma**  **Principal, NIE, Mysore** |
| **Viva-Voce:** | |  | |
| **Name of the Examiners**  **1.**  **2.** | | **Signature with Date** | |

**DECLARATION**

We, Abhishek Singh bearing USN: 4NI19IS003, Abhishek Singh bearing USN: 4NI19IS004, Mohit Kumar bearing USN: 4NI19IS049, R Naveen Kumar bearing USN: 4NI19IS071 students of 8th semester of UG program, Department of Information Science and Engineering, The National Institute of Engineering, Mysuru hereby declare that the project work entitled “DeFi Pay: A futuristic Payment System” has been carried out by us under the guidance of Mr. Suhaas K P, Assistant Professor, Department of I.S.&E.

This project work is submitted to **The National Institute of Engineering**, Mysuru, (An Autonomous institute under VTU, Belagavi) in partial fulfilment of the **Major Project phase-2** requirements for the award of degree in Information Science & Engineering during the academic year 2022-2023. This written submission represents a record of original work and We have adequately cited and referenced the original sources.

Further the matter embodied in this thesis has not been submitted to any other University or Institution for the award of any degree.

Place: Mysuru

Date:

|  |  |
| --- | --- |
|  | (Signature of the students) |
| 1. | Abhishek Shankar |
| 2. | Abhishek Singh |
| 3. | Mohit Kumar |
| 4. | R Naveen Kumar |

**ACKNOWLEDGEMENT**

We are extremely thankful to Dr. Rohini Nagapadma, Principal, NIE, Mysuru, for providing us the academic ambiance and laboratory facilities to work, and everlasting motivation to carry out this work and shape our careers.

We express our sincere gratitude to Dr. Girish, Professor and Head, Dept. of Information Science & Engineering, NIE, Mysuru, for her stimulating guidance, continuous encouragement, and motivation throughout the course of the present work.

We extend our gratitude to our Project-Coordinator Ms. Prajakta M, Assistant Professor the Guide Mr. Suhaas K P, Assistant Professor and Co-Guide Mrs. Nandini B M, Associate Professor Dept. of Information Science and Engineering, NIE, Mysuru, for their support and guidance over the entire course of work.

We take this opportunity to thank all our friends, classmates who always stood by us in difficult situations also helped us in some technical aspects, and last but not least, we wish to express our inspiration and sense of gratitude to parents who were a constant source of encouragement and stood by us as a pillar of strength for completing this work and course successfully.

|  |  |
| --- | --- |
| Abhishek Shankar | 4NI19IS003 |
| Abhishek Singh | 4NI19IS004 |
| Mohit Kumar | 4NI19IS049 |
| R Naveen Kumar | 4NI19IS071 |

**ABSTRACT**

DeFi Pay is a decentralized financial application which is in trend for saving lending and trading the cryptocurrency around the world based on blockchain (Ethereum). In DeFi Pay user lend money to borrower which is collateralized, and borrower invest that money and earn interest from that and later pay back according to smart contract. User also can save their money in pool and earn interest from it. They need to deposit two tokens one is stable and other is volatile so that they can manage loss from one another. We will achieve the above by building a website to interact with the user and we will implement smart contract to which both parties will agree on.

People often wait for long time for financial institution or CeFi to approve loan and transactions apart from charging huge processing fee for the same however with block chain technology the role of intermediary is eliminated which reduces the approval time less than 24 hour. The lender and borrowers can tack the entire history of transactions so there very less chances for financial forgery. Blockchain technology also reduces dependencies on physical documentation which often leads to fraudulent while smart contract cuts service and administration cost. As we step into the future there is a great need of to renovate and upgrade the centralized banking system and pay way for futuristic secure time saving reliable and transparent way for taking and lending credit. This DeFi Pay tunes with current technology and trends of blockchain which ever huge potential to bring about a change in current credit industry.

**Objective:**

1. To create a decentralized system of lending, saving, and trading cryptocurrency in secure manner using blockchain technology.

2. To make DeFi system more transparent, fast, and reliable by improving smart contract.

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
|  | **Chapter Name** | **Page No.** |
|  | **1. List of Figures**  **2. List of Tables** | i  ii |
|  |  |  |
| **Chapter 1** | **Introduction**  1.1 Fundraising  1.2 Introduction to Blockchain  1.3 Structure of Blockchain  1.4 Blocks  1.5 Block Time  1.6 Ethereum  1.7 Smart Contract  1.8 Metamask Chrome Extension  1.9 Rinkeby Test Network  1.10 Web3.js Library | **01**  01  01  02  02  03  03  04  04  04  05 |
|  |  |  |
| **Chapter 2** | **Literature Survey**  2.1 Platform for tracking donations of charitable foundations based on Blockchain Technology  2.2 Proposed solution for trackable donations using blockchain  2.3 Venturing crowdfunding using smart contracts in blockchain  2.4 Crowdfunding Fraud Prevention using Blockchain | **06**  06  06  06  06 |
|  |  |  |
| **Chapter 3** | **System Requirements**  3.1 Hardware Requirements  3.2 Software Requirements | **08**  08  08 |
|  |  |  |
| **Chapter 4** | **System Design**  4.1 Existing System  4.2 Proposed System  4.3 Overview of the System  4.4 Use Case Diagram  4.5 Sequence Interaction in a System  4.6 Data Flow Diagram | **09**  09  10  13  13  14  15 |
|  |  |  |
| **Chapter 5** | **System Implementation**  5.1 Connect Wallet  5.2 Creating a Campaign  5.3 Algorithm to Create a New Campaign  5.4 Contributing to a Campaign  5.5 Algorithm to Donate to an Existing System  5.6 Making a Withdrawal Request  5.7 Algorithm to Create a Withdrawal Request  5.8 Algorithm to Approve the Withdrawal Request | **18**  18  19  19  20  20  20  21  21 |
|  |  |  |
| **Chapter 6** | **Deployment** | **22** |
|  |  |  |
| **Chapter 7** | **Testing**  7.1 Levels of Testing  7.2 Test Cases | **23**  23  24 |
|  |  |  |
| **Chapter 8** | **Cost Effectiveness** | **26** |
|  |  |  |
|  | **Conclusion**  **References** | **27**  **28** |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Fig No.** | **Description** | **Page No.** |
|  |  |  |
| 1.1 | Structure of Blockchain | 02 |
| 1.2 | Blocks in Blockchain | 03 |
| 4.1 | Existing System of Fundraising | 09 |
| 4.2 | Smart Contract system ensures money spent is in control of contributors | 11 |
| 4.3 | Smart Contract | 12 |
| 4.4 | Overview of System Architecture | 13 |
| 4.5 | Use Case Diagram | 14 |
| 4.6 | Sequence Diagram | 15 |
| 4.7 | Creating or Contributing to a campaign | 16 |
| 4.8 | Flowchart of Project Manager | 17 |
| 5.1 | Connecting the Wallet | 18 |
| 5.2 | Creating a New Campaign | 19 |
| 5.3 | Contributing Funds to the Campaign | 20 |
| 5.4 | Withdrawal Request Page | 21 |

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| **Table No.** | **Description** | **Page No.** |
|  |  |  |
| 7.1 | Table of Test Cases | 24 |

**CHAPTER 1**

**INTRODUCTION**

**1.1 DeFi**

DeFi refers to financial applications built on blockchain technologies, typically using smart contracts. Smart contracts are automated enforceable agreements that do not need intermediaries to execute. Anyone with an internet connection can access them to perform financial transactions and many other activities. DeFi consists of applications and peer-to-peer protocols developed on decentralized blockchain networks that require no access rights. The decentralized apps (dApps) are used for easy lending, borrowing, or trading of financial tools. Most DeFi applications today are built using the Ethereum network, but many alternative public networks are emerging that deliver superior speed, scalability, security, and lower costs. The blocks are "chained" together through the information in each proceeding block, giving it the name blockchain. Information in previous blocks cannot be changed without affecting the following blocks, so there is no way to alter a blockchain. This concept, along with other security protocols, provides the secure nature of a blockchain.

**1.2 Introduction to Blockchain**

Blockchain Technology is becoming increasingly evident in our day to day lives. Increased companies have started using blockchain as an underlying network for their daily transactions. When blockchain became popular, it was only used for handling payments using the cryptocurrency Bitcoin, but over the years, numerous studies have been carried upon and have started to suggest that blockchain can be used in many more areas. In blockchain, there is no need for a central authority to approve of these transactions or execute the operations. Blockchain is a peer-to-peer based network which is run by all the nodes that are participating. Because of this, not only Ethereum has a zero-trust network but also all the structural information is kept within the blockchain network. The devices connected to the network are called nodes and they keep a copy of the blockchain and all of the transaction history in the form of a distributed ledger. Devices and built-in sensors are connected to the Internet of Things item, which collects data from different devices and uses statistics to share the most important information with applications designed to address specific needs.

Blockchain is a system of recording information in a way that makes it difficult or impossible to change, hack, or cheat the system. A blockchain is essentially a digital ledger of transactions that is duplicated and distributed across the entire network of computer systems on the blockchain. The goal of blockchain is to allow digital information to be recorded and distributed, but not edited. Blockchain technology was first outlined in 1991 by Stuart Haber and W. Scott Stornetta, two researchers who wanted to implement a system where document timestamps could not be tampered with. Blockchain enables 100% transparency and ensures transactional integrity and non-repudiation across a distributed ledger.

Blockchain is being used in various walks of life, companies have started to use it in

various new ways such as personality identity, prevent fraud money laundering scams etc.

**1.3 Blocks**

Blocks hold batches of valid transactions that are hashed and encoded into a Merkle tree. Each block includes the cryptographic hash of the prior block in the blockchain, linking the two. The linked blocks form a chain. This iterative process confirms the integrity of the previous block, all the way back to the initial block, which is known as the genesis block.

Sometimes separate blocks can be produced concurrently, creating a temporary fork. In addition to a secure hash-based history, any blockchain has a specified algorithm for scoring different versions of the history so that one with a higher score can be selected over others. Blocks not selected for inclusion in the chain are called orphan blocks. Peers supporting the database have different versions of the history from time to time. They keep only the highest- scoring version of the database known to them.

Whenever a peer receives a higher-scoring version (usually the old version with a single new block added) they extend or overwrite their own database and retransmit the improvement to their peers. Blockchains are typically built to add the score of new blocks onto old blocks and are given incentives to extend with new blocks rather than overwrite old blocks. Therefore, the probability of an entry becoming superseded decreases exponentially as more blocks are built on top of it, eventually becoming very low.

**1.6 Ethereum**

Ethereum blockchain allows us to execute code with the Ethereum Virtual Machine (EVM) on the blockchain with something called a smart contract. Smart contracts are where all the business logic of the application lives. Ethereum is a public blockchain which is decentralized framework and is completely independent and is not constrained by anybody by any means. Then it can be incorporated in the Ethereum blockchain which cannot be altered by any individual. A smart contract is a computer protocol that allows us to facilitate and verify the performance of a contract. These transactions are trackable and irreversible.

**1.7 Smart Contract**

Smart contracts are in charge of reading and writing data to the blockchain, as well as executing business logic. Smart contacts are written in a programming language called Solidity, which looks a lot like JavaScript. It is a full-blown programming language that will allow us to do many of the same types of things JavaScript is capable of, but it behaves a bit differently because of its use case.

Project will have a traditional front-end client that is written in HTML, CSS, and JavaScript. Instead of talking to a back-end server, this client will connect to a local Ethereum blockchain that need to be installed. The business logic about the Dapp is coded in a campaign smart contract with the Solidity programming language. Smart contract is deployed to the local Ethereum blockchain and allow accounts to start campaigning.

System architecture is a model that defines the behavior of a system in the conceptual model. The huge systems are decomposed into subordinate systems to provide a similar set of services. The beginning layout strategy of perceiving these sub-systems and building up a structure for sub-systems control and cooperation is called architecture design.

**1.8 Metamask Chrome Extension**

In order to use the blockchain, app must connect to it (block chain is a network). A special browser extension has to be installed in order to use the Ethereum block chain. That’s where metamask comes in. It will be able to connect to the local Ethereum blockchain with the personal account and interact with the smart contract.

**1.9 Rikeby Test Network**

The Rinkeby testnet is an Ethereum testnet that developers use to test and perfect their own decentralized applications. The network is run by pre-authorized nodes, which prevents spam attacks and increases performance. Developers can use a Rinkeby Faucet to get free testnet ETH and test their smart contracts without the risk of losing real financial assets. The Rinkeby testnet is mainly used for blockchain testing before deployment on the Ethereum mainnet. It is a fork of the Ethereum mainnet.

**1.10 Web3.js Library**

Digital assets such as cryptocurrencies (or programmable tokens) and smart contracts are a central component of decentralized applications (DApps), in that they are deployed on the blockchain. However, to interact with these on-chain components, transactions need to be created on the blockchain. For a user or off-chain software to create a transaction on the blockchain, a node needs to relay the transaction to the underlying peer-to-peer (P2P) network. Web3.js is a collection of libraries that allows programmers to interact with these on-chain components, by being able to facilitate a connection to Ethereum nodes.

**1.11 Truffle**

Truffle is a popular development framework and suite of tools specifically designed for building and deploying smart contracts on the Ethereum blockchain. It provides a comprehensive environment that simplifies the entire development process for Ethereum-based decentralized applications (DApps). Truffle offers features such as automated contract compilation, testing, and deployment, along with built-in support for smart contract management and migration. Developers can write their contracts using Solidity, the programming language for Ethereum smart contracts, and leverage Truffle's powerful command-line interface and development environment. With its robust tooling and extensive library of plugins, Truffle has become a go-to solution for developers seeking efficiency and productivity when working on Ethereum-based projects.

**1.12 Ganache**

Ganache is a personal blockchain emulator and development tool provided by Truffle Suite. It allows developers to create and manage local Ethereum networks for testing and development purposes. With Ganache, developers can simulate the behavior of an actual Ethereum network without the need for real Ether or interacting with the main Ethereum network. It provides a user-friendly interface to quickly set up and configure blockchain networks, complete with accounts, private keys, and preloaded Ether for testing smart contracts. Ganache offers powerful features like gas customization, transaction logging, and blockchain visualization, making it an invaluable tool for smart contract development, debugging, and experimentation.

**1.13 Solidity**

Solidity is a programming language specifically designed for writing smart contracts on the Ethereum blockchain. It provides developers with a user-friendly syntax, resembling JavaScript, making it accessible and easy to learn. Solidity allows the creation of secure and auditable smart contracts, with features like data structures, functions, and modifiers. It supports inheritance and includes mechanisms to prevent common vulnerabilities, ensuring the reliability and security of contracts. Solidity contracts are compiled into bytecode and executed on the Ethereum Virtual Machine (EVM). With its rich features and compatibility with Ethereum's ecosystem, Solidity has become the go-to language for building decentralized applications, decentralized finance (DeFi) protocols, and various other blockchain-based solutions.

**CHAPTER 2**

**Existing System & Proposed System**

**2.1 Existing System**

Now currently existing system in world is centralized finance. In which there is a centralized authority (like RBI in India) which monitor, regulate the money and all related monetary policies. It is highest authority under which many branch authority like various banks, Mutual grants, bond grants etc functions.

In centralized finance, money is held by banks and third parties who facilitate money movement between parties, with each charging fees for using their services. A credit card charge starts from the merchant and moves to an acquiring bank, which forwards the card details to the credit card network. The network clears the charge and requests a payment from the bank. Each entity in the chain receives payment for its services, generally because merchants must pay for the use of credit and debit cards. All financial transactions are overseen in centralized finance, from loan applications to a local bank's services. The process of centralization refers to the concentration of planning and decision-making processes inside an organization to a single leader or location. In a centralized organization, the head office retains decision-making authority, while all subordinate offices receive orders from the main office. The head office houses the executives and specialists who make crucial decisions.

**2.2 Proposed System**

We are creating a decentralize finance Ethereum blockchain based website. Decentralized finance eliminates intermediaries by allowing people, merchants, and businesses to conduct financial transactions through emerging technology. It leverages smart contracts, decentralized applications (DApps), and blockchain platforms to enable various financial activities in a decentralized and transparent manner. Through peer-to-peer financial networks, DeFi uses security protocols, connectivity, software, and hardware advancements. Wherever there is an internet connection, individuals can lend, trade, and borrow using software that records and verifies financial actions in distributed financial databases. A distributed database is accessible across various locations as it collects and aggregates data from all users and uses a consensus mechanism to verify it. DeFi systems are designed to be open and permissionless, meaning that anyone with an internet connection can participate and access the services which means there are gate keeper.

**CHAPTER 3**

**LITERATURE SURVEY**

**[1]: Decentralized finance (DeFi): The Future of Finance and Defi Application for Ethereum blockchain based Finance Market B. Sriman; S. Ganesh Kumar (Published in: 2022 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI)).**

• DeFi is totally about reinverting traditional financial services such as borrowing, lending and trading built using decentralized infrastructure like public blockchain and smart contracts.

• Using the article, we are building a decentralized financial application to exchange your tokens, lending, borrowing of the any cryptocurrency without any of the problems to an existing system and to enable safe positive yielding to stored assets.

• This article examined how to work transactions that are stored in the digital ledgers are recorded in the Blockchain.

• It stores all the valuable data about the user's transactions and using this article we are able to understand the concept of decentralised finance system how it is to be used in project.

**[2]: Blockchain Akanksha Kaushik; Archana Choudhary; Chinmay Ektare; Deepti Thomas; Syed Akram (Published in: 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT))**

• The concept of bitcoin is in a way difficult to theorize but can be implemented in practice.

• Bitcoin has its own perks when it comes in comparison to the traditional old bank

transactions. It is a decentralized form of currency; in simple terms it means that nobody.

rules over it.

• The presence of many redundant copies of the transaction database eliminates any third-party rule over the money you own and lets you exercise total control over it, the

government can't freeze your money.

• This article helped us to understand the basic use cases of blockchain and its future

**CHAPTER 3**

**SYSTEM REQUIREMENTS**

**3.1 Hardware Requirements**

The following are the Hardware Requirements for the running the system

Hard Disk : 50 GB or more

Ram : 8 GB Recommended

Processor : i3 / AMD Fx 4100 (or greater)

Processor Speed : 1.5 GHz or higher

**3.2 Software Requirements**

Operating System : Windows / Ubuntu / Mac

Languages : HTML, CSS, JavaScript, and Solidity

Tools : VS code, Web Browser (Google Chrome)

: Rinkeby Test Network

: Web3.js, Truffle, Gana

: Metamask Chrome Extension

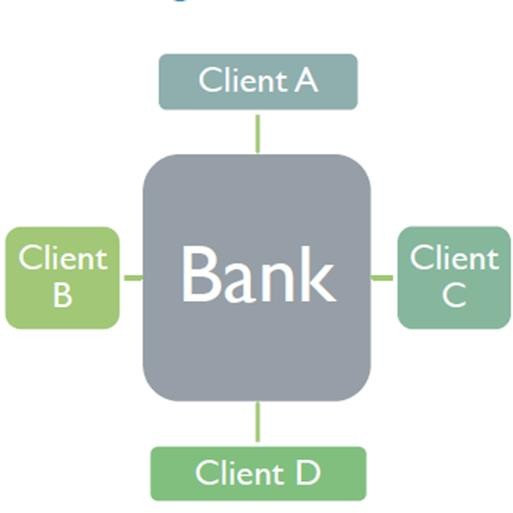
**CHAPTER 4**

**SYSTEM DESIGN**

**4.1 Existing System**

Now currently existing system in world is centralized finance. In which there is a centralized authority (like RBI in India) which monitor, regulate the money and all related monetary policies. It is highest authority under which many branch authority like various banks, Mutual grants, bond grants etc functions.

In centralized finance, money is held by banks and third parties who facilitate money movement between parties, with each charging fees for using their services. A credit card charge starts from the merchant and moves to an acquiring bank, which forwards the card details to the credit card network. The network clears the charge and requests a payment from the bank. Each entity in the chain receives payment for its services, generally because merchants must pay for the use of credit and debit cards. All financial transactions are overseen in centralized finance, from loan applications to a local bank's services. The process of centralization refers to the concentration of planning and decision-making processes inside an organization to a single leader or location. In a centralized organization, the head office retains decision-making authority, while all subordinate offices receive orders from the main office. The head office houses the executives and specialists who make crucial decisions.



**Fig 4.1** Existing System of Fundraising

**4.1.1 Disadvantages of Existing system**

* **Fees:** Most crowdfunding platforms will take a percentage of the contributions raise. These fees are minimal but may still reduce the amount of money you’d otherwise get.
* **Platform risk**: Bankruptcy of a platform can directly adversely affect both entrepreneurs and investors.
* **Fraud by a crowdfunding platform**: Due to the relatively new and growing market, not many systems have yet been designed to prevent or compensate for such fraud.
* **Fraud by borrowers**: It is possible that borrowers set up a fraudulent campaign. This is because there is less strict control than with other financing methods. If you decide to invest in such projects, there are fewer safety nets to absorb your losses.

A major disadvantage of crowdfunding via crowdfunding platforms is that if you fail to collect the budget you have set in time, your project will be removed from the platform, and you will have to start again from scratch. In addition, in the event of bankruptcy, there may be a claim against your personal assets. Some platforms offer the option to re-promote a revised version of the campaign, but this is not a general right.

**4.2 Proposed System**

We are creating a decentralize finance Ethereum blockchain based website. Decentralized finance eliminates intermediaries by allowing people, merchants, and businesses to conduct financial transactions through emerging technology. It leverages smart contracts, decentralized applications (DApps), and blockchain platforms to enable various financial activities in a decentralized and transparent manner. Through peer-to-peer financial networks, DeFi uses security protocols, connectivity, software, and hardware advancements. Wherever there is an internet connection, individuals can lend, trade, and borrow using software that records and verifies financial actions in distributed financial databases. A distributed database is accessible across various locations as it collects and aggregates data from all users and uses a consensus mechanism to verify it. DeFi systems are designed to be open and permissionless, meaning that anyone with an internet connection can participate and access the services which means there are gate keeper.

**4.2.1 Advantages of Proposed system**

* Startups are not going to rely on any platform or combination of platforms to enable creators to raise funds. Startups no longer be beholden to the rules, regulations, and whims of the most popular crowdfunding platforms on the internet.
* It also eliminates the problem of fees, while blockchain upkeep does cost a bit of money, it will cut back drastically on transaction fees. This makes crowdfunding less expensive for creators and investors.
* Any project using a blockchain based crowdfunding model can potentially get funded. Also, any person with an internet connection can contribute to those projects.
* Blockchain based crowd funders wouldn’t have to worry about the “Fraud” that have plagued modern day crowdfunding projects. Instead, contributors will immediately receive fractional enterprise or product ownership.

**4.3 Overview of the System**

Diagram

Description automatically generated For the implementation, Ethereum Blockchain is used because it is one of the widely used and open-source platforms for developing and deploying decentralized applications. Ethereum provides a wide range of services and solutions with its readily available development tools and smart contracts. A smart contract is a self-executing program that runs on the blockchain. In order for it to get compiled and deployed on the blockchain, each node on the network then executes the contract in exchange for some ether. The currency required to execute a contract is called gas amount and varies from contract to contract.

**Fig 4.4** Overview of System Architecture

**4.4 Use Case Diagram**

A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has. Here the use cases are represented by either circles or ellipses. The actors are shown as stick figures.

Use case diagrams are also called as high-level design diagrams. The use cases and actors in use-case diagrams describe what the system does and how the actors use it, but not how the system operates internally. Use-case diagrams illustrate and define the context and requirements of an entire system and the important parts of the system.

This UML Use case diagram shows the options or the interface available for the managers and the donors. Managers have the option to Create Campaign, Add Details, Create Spend Request. Donors have options to donate and vote for the campaign. Backing a project, Viewing the status of the project, Reviewing the available Project options are available for Diagram

Description automatically generatedboth the managers and the donors.

**Fig 4.5:** Use Case Diagram

**4.5 Sequence Interaction in a System**

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the logical view of the system under development. Sequence diagrams are sometimes called event diagrams or event scenarios.

Here the parallel vertical lines show different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur.

Here in the below Sequence diagram the event or a project starts by creation of the project, the create project request is sent from the beneficiary to the smart contract through the platform. Then the success from the smart contract is returned to the platform. After the creation the Project manager uploads the features and outcomes of the project based on that donor donate. The donated funds will be in the smart contract. When the manager wants to use the money, he has to request the money saying where it will be used. Now to allow the control over the donor’s money the donors have a voting chance to agree/not agree for the use Diagram

Description automatically generatedof money. If the votes are more than 50% funds are transferred.

**Fig 4.6** Sequence Diagram

**4.6 Data Flow Diagram**

A control-flow diagram can consist of a subdivision to show sequential steps, with if-then-else conditions, repetition, and/or case conditions. Suitably annotated geometrical figures are used to represent operations, data, or equipment, and arrows are used to indicate the sequential flow from one to another.

Project manager creates new project by mentioning the name of the project, the description of the project and the minimum contribution to that project. And the contributors then can view the all the open projects in crowd funding platform and can choose any project for which they want to contribute. To mark themselves as contributors, they have to invest minimum contribution for that project which project manager has mentioned while creating the project. And this money is added to the wallet which can be used by the project mangers

#### Diagram Description automatically generated

**Fig 4.7** Creating or Contributing to a campaign

It is a program which is written in solidity language to handle all the transactions automatically. As shown in Fig. 3, the project manager has to first create the project by mentioning the name, description, and the minimum contribution for that project. Then user can create the spending request for spending the money contributed by the investors. For this project creators have to mention the description about where they are going to spend the money, the amount of money they are going to spend and the address of the vendor who will provide some service. If more than half of the investors agree for the spending request, then the project manager can send the money to vendor’s address. Then the vendor provides the service requested by the project manager.

Diagram

Description automatically generated**Fig 4.8** Flowchart of Project Manager

**CHAPTER 5**

**SYSTEM IMPLEMENTATION**

**5.1 Connect Wallet**

In order to perform any transactions, be it creation of a campaign or contributing to one, a user first needs to connect an Ethereum wallet to the site. We have made use of a browser extension called Metamask which is a software cryptocurrency wallet used to interact with the Ethereum blockchain. Allows users to access their Ethereum wallet via a browser extension or mobile application, which may be used to communicate with country-divided applications. MetaMask was developed by ConsenSys Software Inc., a blockchain software company focused on Ethereum-based tools and infrastructure.

Graphical user interface, application

Description automatically generated

**Fig 5.1** Connecting the Wallet

**5.2 Creating a Campaign**

**A screenshot of a computer

Description automatically generated**Just like Crowdfunding in the real world as well as on other crowdfunding platforms, anyone can create a campaign in a few minutes. The campaign information will be managed by the Ethereum-based smart contract and thus cannot be tampered with.

**Fig 5.2** Creating a New Campaign

**5.3 Algorithm to Create a New Campaign**

Step 1: Start

Step 2: Click on Connect Wallet Button

Step 3: Click on Create Campaign Button

Step 4: Enter the details of the Campaign

Step 5: Click on Create Button to create a new Campaign

Step 6: Approve the Campaign through the metamask Wallet

Step 7: Stop

**5.4 Contributing to a Campaign**

A screenshot of a computer

Description automatically generated with medium confidenceOnce a campaign has been created, users can share the campaign, and anybody can contribute to the campaign. The funds will go to the address of the campaign and not to the creator of the campaign, thus making the process more efficient and anti-fraudulent. The donors can donate their money if they are pleased with the campaigns and all the money will be present in the smart contract making the fundraising transparent.

**Fig 5.3** Contributing Funds to the Campaign

**5.5 Algorithm to Donate to an Existing Campaign**

Step 1: Start

Step 2: Select an existing campaign on the home page

Step 3: If interested donate to the Campaign

Step 4: Input the donation greater than the minimum amount

Step 5: Click on Contribute

Step 6: Approve the donation through metamask Wallet

Step 7: Stop

**5.6 Making a Withdrawal Request**

If you are the creator of a fund, you might need to withdraw from the available funds for various reasons. You can create a Withdrawal Request by the flow given below, which must be approved by the majority of approvers.

If you are a Contributor who has contributed more than the Minimum Contribution (specified in the campaign), then you are an approver. You can vote on the Withdrawal requests made by the creator, and either approve or deny the request.

A screenshot of a computer

Description automatically generated with medium confidenceNo funds can be withdrawn without the approval of at least 50% of the approvers. This provides complete transparency in the process of withdrawing funds. Approvers can see where their money is going. The creator is not the intermediary of the transfer of funds.

**Fig 5.4** Withdrawal Request Page

**5.7 Algorithm to Create a withdrawal Request**

Step 1: Start

Step 2: Select an existing campaign on the home page

Step 3: Click on the view withdrawal request

Step 4: Enter the details for withdrawal

Step 5: Create a ticket to run through the smart contract to get approved

Step 6: Stop

**5.8 Algorithm to Approve the Withdrawal Request**

Step 1: Start

Step 2: Select an existing campaign on the home page

Step 3: Click on view withdrawal Request

Step 4: Click on Approve Button

Step 5: Stop

**CHAPTER 6**

**DEPLOYMENT**

Deployment encompasses all the processes involved in getting new software or hardware up and running properly in its environment, including installation, configuration, running, testing, and making necessary changes.

The below mentioned are the steps to deploy the application:

* First install all the required dependencies like Package manager, and create account in Infura, Metamask.
* Use the Metamask chrome extension to create accounts on rinkeby network
* Use the link <https://faucets.chain.link/>, to add test Ether to your account.
* Start writing the smart contract it contains all the business logic of dApp. It will be in charge reading from and writing to the Ethereum blockchain.
* Migrate smart contract to the rinkeby blockchain.
* Type the various input needed for the project like campaigners, managers, doners etc.
* Solidarity Mapping is used to store the approvers.
* Compile the Contract using the node compile.js
* Deploy Contract by going into smart-contract Directory and run. node deploy.js
* Copy the contract deploy address and replace it in factory.js file.
* Add your "infura end point link" in web3.js file
* Test the application, make sure that all the tests pass.
* Next, build the client-side application by using HTML and JavaScript. Set up JavaScript libraries, initialize contacts, lay out all the content on the page with the data from the smart contract.
* Cast the votes in the election and test the voting function if the function increments the vote count for the candidate and test that the voter is added to mapping whenever they vote.
* Finally, run the application.

**CHAPTER 7**

**TESTING**

Testing is an essential footstep in the development of a system. Testing is the course of action of verifying the correctness of the system that is already developed. After the system is developed, it needs to be verified with the functionalities that each functional specification is working in a correct manner. It also ensures the newly developed system meets the quality requirement and genuineness. Testing is performed at various levels of the system with the sole aim of making a secure and qualified system. In other words, the testing is performed to check whether the system is working as in the same way the system was designed and expected to work. The testing is mainly performed to achieve and affirm the quality of the project. The testing performs the quality assurance for the software. Testing involves writing test cases for each operation to check and verify the functionality of the module.

**7.1 Levels of Testing**

**7.1.1 Unit Testing**

Unit testing is the method in which individual units of source code or modules are tested. These discrete units are assessed for their functionality. The unit testing is also called “Module Testing”. The unit testing makes use of specific paths in the control structure of the module that ensures complete testing of the module. The expected output is validated against the actual output and the module functionality is inferred.

**7.1.2 Integration Testing**

The integration testing involves integrating of several modules and testing for the functionality in the combined effort. The integration testing helps in systematic technique for constructing the program structure. When the modules are integrated, certain classes may lead to inconsistencies. These inconsistencies affect both the modules and their functionality. To deal with such issues, the integration testing is performed.

**7.1.3 System Testing**

The system testing involves ensuring the fully integrated system meets the desired requirements. It is used to check the system compliance to meet the requirements. The system testing does not require the knowledge of inner logic and its design.

**7.1.4 Acceptance Testing**

The acceptance testing is mainly carried out to find out whether the requirements and specifications are met. It involves performing tests on the system. The acceptance testing requires significant participation by the end user. It also determines whether the functional requirements are met or not.

**7.1.5 Performance Testing**

The performance testing is the testing performed to find out how the developed system reacts in ways of stability and responsiveness. The performance testing involves load test, stress testing, soak testing and isolation/segregation testing

**7.2 Test Cases**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Testcase name | Operation | Expected result | Actual Result | **Pass/Fail** |
| Run yarn package manager | Launches the website and then starts the server from the terminal | launch the website and start the server | Website successfully launched and server started | Pass |
| Links accessibility | Verify the links are clickable in the homepage | Successful if taken to the clicked link page | Successfully taken to the specified page | Pass |
| Form fields accept Data for entry | Verifying the form fields are accepting data from the user | Data should be entered in the form fields | Data can be successfully entered in the form fields | Pass |
| Submit the data button in the form | Form data should be submitted on the click and navigate to respective page | Successful if entry is valid show the respective details | Successfully takes to respective page | Pass |
| Accessing metamask through Chrome extension | Opens the Metamask application | Opens the Extension for chrome | Extension opened successfully | Pass |
| Job Module Import rinkeby account of user to Metamask | Importing the rinkeby account of the user to the metamask extension | Account added successfully to the metamask | Successfully added account to metamask | Pass |
| Job Module- Web3.js Deployment | Deploys the smart contract to rinkeby test network | Successful if deployed | Deployed the contract successfully | Pass |
| Testcase name | Operation | Expected result | Actual Result | Pass/Fail |
| Solidity Mapping | Place the actors’ details in the blockchain | Successful if store all the actor’s info | Successfully Stored | Pass |
| Addition of campaigns | Add the campaigns to blockchain using Ethereum | Successful if add the campaign | Successfully added | Pass |
| Addition of donors after donation | Add the doners to blockchain | Successful if add the doner | Successfully added | Pass |
| Doners Login | Enter the login credentials with respective metamask | Successful if verified and logged in | Successfully logged in | Pass |
| Metamask confirmation | Verify the account, gas limit, available balance, and perform the | Successful if respective verification is done | Successfully verified | Pass |
| Doners tracking the product | Doners tracking the campaigns and campaign related funds | Successful if the related information is found | Successfully verified | Pass |

**Table 7.1** Table of Test Cases

**CHAPTER 8**

**COST EFFECTIVENESS**

Our proposed system is a lot more cost effective than the traditional campaigning approach in the following ways:

* The cost per campaign which was more due to the presence of middlemen can be cut down.
* Distributed ledger technology can reduce the cost of financial services infrastructure, provide an opportunity to streamline the assets and infrastructure and significantly reduce IT costs.
* Saves time and money in search of donors which could be easily available on the platform

**CONCLUSION**

Finally, it is concluded that the crowdfunding using blockchain is a relatively new concept and it’s still in exploratory stage where numerous legal issues and specialized issues need to be handled.

Blockchain does have the potential to become an integral part of the operation of many businesses, offering scalability, security, and computing power at a lower CAPEX and OPEX. But of course, as is the case with most new technology service offerings, there are a number of risk-based issues that need to be carefully considered before business, particularly heavily regulated ones, can start to fully realize the potential benefit.

**REFERENCES**

**IEEE Papers**

[1] H. Saleh, S. Avdoshin and A. Dzhonov, "Platform for Tracking Donations of Charitable Foundations Based on Blockchain Technology," 2019 Actual Problems of Systems and

Software Engineering (APSSE), 2019, pp. 182-187, doi: 10.1109/APSSE47353.2019.00031.

[2] N. S. Sirisha, T. Agarwal, R. Monde, R. Yadav, and R. Hande, "Proposed Solution for Trackable Donations using Blockchain," 2019 International Conference on Nascent

Technologies in Engineering (ICNTE), 2019, pp. 1-5, doi:10.1109/ICNTE44896.2019.8946019.

[3] N. Yadav and S. V., "Venturing Crowdfunding using Smart Contracts in Blockchain," 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT), 2020, pp. 192-197, doi: 10.1109/ICSSIT48917.2020.9214295.

[4] S. Pandey, S. Goel, S. Bansla and D. Pandey, "Crowdfunding Fraud Prevention using Blockchain," 2019 6th International Conference on Computing for Sustainable Global Development (INDIACom), 2019, pp. 1028-1034.

[5] Blockchain & Smart Contracts: <https://www.dappuniversity.com/articles/how-to-build-a-blockchain-app>

[6] web3.js - Ethereum JavaScript API: <https://web3js.readthedocs.io/en/v1.3.4/>

[7] Metamask Ethereum Wallet: <https://metamask.io/>

[8] Rinkeby Ethereum Test Network: <https://www.rinkeby.io/#stats>