Visvesvaraya Technological University,Belagavi–590018

“ Jnana Sangama ”, Belagavi – 590018

A PROJECT REPORT ON

“CROWDFUNDING PLATFORM USING BLOCKCHAIN TECHNOLOGY”

Submitted in partial fulfilment of the requirements for the award of

BACHELOR OF ENGINEERING IN INFORMATION SCIENCE AND ENGINEERING

For the Academic year 2022-2023

Submitted by:

ANANYA	**4SH19IS001**
NIDHISHREE	**4SH19IS004**
SHREYA	**4SH19IS009**

Under the Guidance of:

Mrs. Nisha Veronica Coutinho

Assistant Professor

Department of CSE

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

SHREE DEVI INSTITUTE OF TECHNOLOGY

KENJAR, MANGALURU-574142

SHREE DEVI INSTITUTE OF TECHNOLOGY

(An Institution under VTU, Belagavi)

MANGALURU-574142

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

CERTIFICATE

Certified that the project work entitled "**CROWD FUNDING PLATFORM USING BLOCKCHAIN TECHNOLOGY**" is a bonafide work carried out by **ANANYA, NIDHISHREE and SHREYA** bearing USN's **4SH19IS001, 4SH19IS004 and 4SH19IS009** respectively in partial fulfilment of eight semester project, regards to the subject "**Major Project"** for the award of degree of **Bachelor of Engineering in Information Science and Engineering** of the Visvesvaraya Technological University, Belagavi during year 2022-2023. It is certified that all corrections/suggestions indicated for 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 degree of Bachelor of Engineering.

Signature of Guide	Signature of HOD	Signature of Principal
Mrs. Nisha Veronica Coutinho	**Prof. Anand S Uppar**	**Dr. K. E. Prakash**
Asst. Professor	**HOD**	**The Principal**
Dept of CSE	**Dept of CSE**	

EXTERNAL VIVA

Name of the Examiners	Signature with Date
1. _____	_____
2. _____	_____

SHREE DEVI INSTITUTE OF TECHNOLOGY

KENJAR, MANGALURU– 574142

Department of Information Science and Engineering

DECLARATION

We **Ananya, Nidhishree and Shreya** bearing USN **4SH19IS001, 4SH19IS004, 4SH19IS009** students of Eighth semester Bachelor of Engineering, Information Science and Engineering, Shree Devi Institute of Technology, Mangalore declare that the project entitled **"CROWDFUNDING PLATFORM USING BLOCKCHAIN TECHNOLOGY"** has been duly executed by us under the guidance of **Mrs. Nisha Veronica Coutinho,** Asst. Professor, Department of Computer Science and Engineering, Shree Devi Institute of Technology, Mangalore and submitted for the requirement for the eighth semester project of **Bachelor of Engineering in Information Science and Engineering** during the year 2022-2023.

Date:
Place: Mangalore

ANANYA [4SH19IS001]
NIDHISHREE [4SH19IS004]
SHREYA [4SH19IS009]

ACKNOWLEDGEMENT

A successful project is a fruitful culmination of the efforts of many people. Some directly involved and others who have quietly encouraged and extended their invaluable support throughout its progress.

We would like to convey my heartfelt thanks to **our Management** for providing the good infrastructure, laboratory facility, qualified and inspiring staff whose guidance was of great help in successful completion of this project.

We are extremely grateful and thankful to our beloved director and principal **Dr. K.E Prakash**, Shree Devi Institute of Technology, Kenjar for providing the congenial atmosphere and necessary facilities for achieving the cherished goal.

With heartiest gratitude, we would like to thank **Prof. Anand S Uppar**, HOD, Department of Computer Science and Engineering for his support, guidance and encouragement.

We are profoundly indebted to our Guide, **Asst. Prof. Nisha Veronica Coutinho**, Department of Computer Science and Engineering, for their guidance throughout the seminar by innumerable acts of timely advice and encouragement.

We also thank all other teaching staff and non-teaching staff for allowing us to carry out the project work.

Finally, we would like to thank our family for their support and understanding, to whom we owe so much.

ANANYA [4SH19IS001]
NIDHISHREE [4SH19IS004]
SHREYA [4SH19IS009]

ABSTRACT

Governments need to cater to a huge number of responsibilities of a state. The working of state governments involves huge number of transactions towards various operations that need to be carried out throughout the state. This includes new projects, repair and maintenance works, awarding contracts, paying of government employees, farmer schemes and so on. A major hurdle that the top government face is the low-level corruption that is sometimes impossible to track which deprives the state progress. Tracking it is a very difficult task due to the current system. Here we propose a smart system to track funds allocated to the state government as they travel through the government process at each stage. We here make use of blockchain technology to secure the transactions at every stage while maintaining transparency in every transaction sealing every transaction with proofs as the funds move ahead. Blockchain, originally block chain, is a growing list of records, called blocks that are linked using cryptography. Each block contains a cryptographic hash of the previous block, a timestamp, and transaction data. In this project researcher use Blockchain Algorithms for security like AES for Encryption and Decryption By design, a blockchain is resistant to modification of the data. In this paper we propose a system to track funds allocated to the government as they travel through the government process at each stage using Key pair generation algorithm, Metadata file decryption and Data verification algorithms. This system uses block-chain technology to maintain the transparency & security at every stage as the funds move ahead. This system allows us to maintain the crystal-clear record with all users who are connected in the chain to transaction the data on a need-to-know basis. The system makes use of encryption to secure transactional data using hash values to maintain a block of transactions in a chain manner, which is maintained & verified by every node involved to verify the transaction and save the data in a transparent form within the government. The system allows for a full proof, secure authentic fund allocation & fund tracking system help to form an incorruptible government procedure.

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1

INTRODUCTION

1.1 Problem Definition

The working of governments involves huge number of transactions towards various operations that need to be carried out throughout the nation. This includes new projects, repair and maintenance works, awarding contracts, paying of government employees, farmer schemes and so on. Usually when a project is allocated funds, there is no knowledge as to how these funds are being used and a large part of it is never show in records due to corruption. To solve this problem, a system has been proposed using Blockchain to provide the transparency.

1.2 Scope and importance

A major obstacle that the top government face is the low-level corruption that is some- times not possible to track which deprives the state progress. Blockchain technology is an upcoming technology and said to be one of the most promising technologies which would revolutionize the world. Building a decentralized database infrastructure using smart contract on the Ethereum blockchain that would solve the problem of current traditional centralized database.

Here we make use of blockchain technology to secure the transactions at every stage while maintaining transparency in every transaction sealing every transaction with proofs as the funds move ahead.

Chapter 2

SOFTWARE REQUIREMENTS SPECIFICATION

2.1 Introduction

A Software Requirement Specification (SRS) is a complete description of the behaviour of the system to be developed. It includes a set of use cases that describes all the interactions that users will have with the software. Use cases are also known as functional requirements. In addition to use cases, the SRS also contain non-functional requirements. The main purpose of System Requirement Specification (SRS) is to translate the ideas in the minds of a client into formal document. Through SRS the client clearly describes what it expects from the proposed system and the developer clearly understand what capabilities are required to build the system.

2.2 Purpose

The purpose if this document is to serve as a guide to developers and testers who are responsible for the development of the system.

2.3 Product Scope

The transactions are the very important factor in handling financial data and tracking the funds effectively. To avoid mishandling of finance transactions must be secure and transparent and it need to be stored safely for future refences. Implementing this using blockchain will provide more convenience and protection.

2.4 External Interface Requirements

This section specifies the user, hardware and software interface requirements. It provides the information required to ensure compatibility between the application and the computer system.

2.4.1 Hardware Interfaces

- Processor: Any processor above 500 MHz
- RAM: 4GB or above
- Input Device: Standard keyboard and Mouse
- Output Device: Monitor

2.4.2 Software Interfaces

- Operating System: Windows 7 or above
- Node.js Gitbash
- Ganache Blockchain
- Metamask

2.5 Functional Requirements

Input: Government Fund allotment details

Technology: Blockchain

Output: To create transparent fund transfer system and prevent corruption

2.6 Non-Functional Requirements

- **Efficiency**
  The system must able to give the correct tracking of government funds.

- **Availability**
  System will be available at any point of time when the system is connected to the internet.

- **Portability**
  This application can be ported to any of the window's system, i.e., for all windows operating system from windows 7 to 10.

- **Maintainability**
  The system can be repaired within a specified period of time. If maintainability is increased repair time will be comparatively less.

- **Reliability**
  The system will function under stated conditions without failure for the given period of time.

Chapter 3

DESIGN

Software design is a stage in the software engineering process at which an executable software system is developed. It is an activity involving identification of software components based on the needs of the client. It deals with representing the client's requirement, as stated in the software requirements specification document, into a format that can be easily implemented using a programming language. The software design phase is the first step in the Software Design Life Cycle (SDLC), which shifts the focus from the problem to the solution.

3.1 Architecture Diagram

A system architecture is the conceptual model that denes the structure, behaviour, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviours of the system. A system architecture can consist of system components and the sub-systems developed, that will work together to implement the overall system. There have been efforts to formalize languages to describe system architecture, collectively these are called architecture description languages (ADLs).

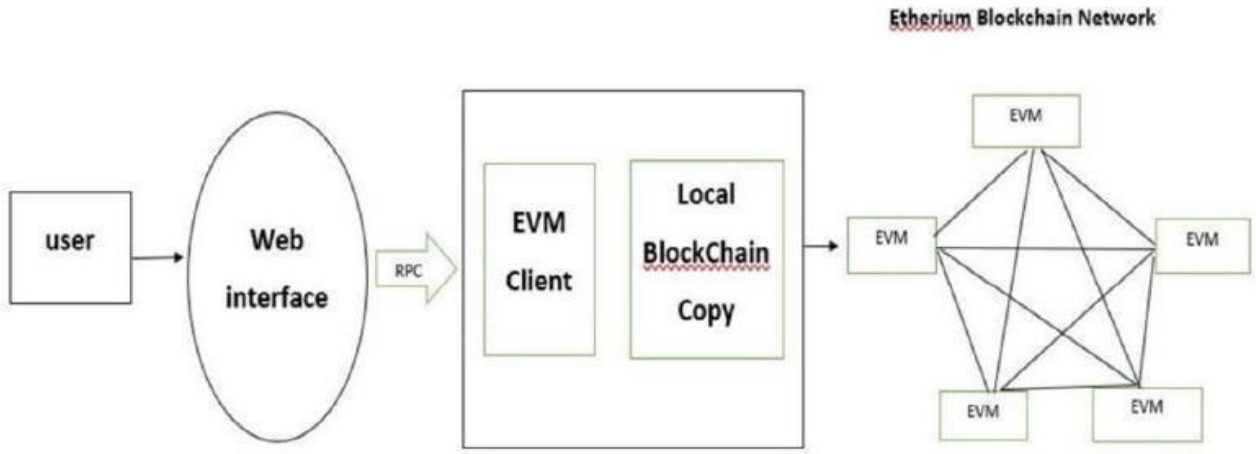
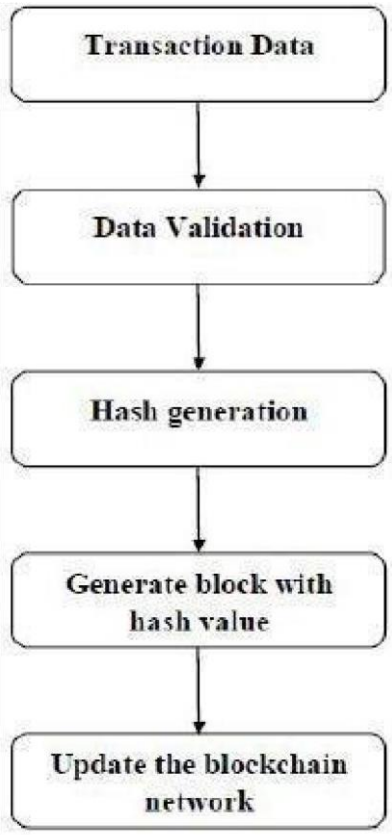


Figure 3.1.1 Architecture Diagram

3.2 Data Flow Diagram

A Data flow diagram or DFD is a method is representing the flow of any data of a particular system or process. It also provides the information regarding the inputs and outputs of every single object and the process itself, as shown in Fig 3.2.1 The DFD is considered to be an abstract structure in creating an overview of the system without elaborating the detail study of the structure. DFDs can

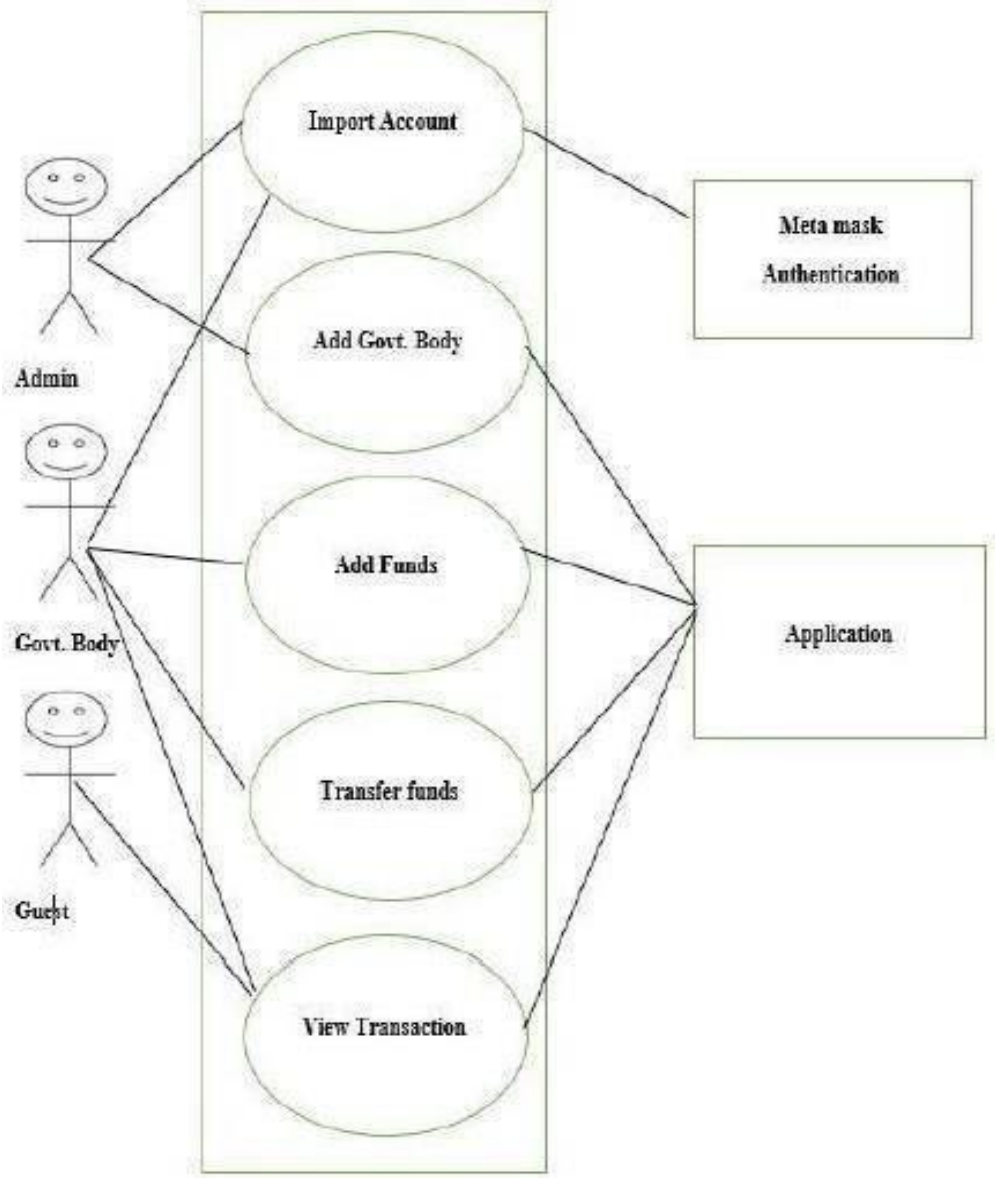
also be for the visualisation of data processing. It is important as it shows the implementation of information given to the system as an input whose purpose is to give out the required output. The data flow diagram does not provide any information regarding the timing or the process ordering or how the process will be processing the data in either a parallel way or a sequence way. A DFD depicts the types of data that will be input to and output from the system, as well as how the data will flow through the system and where it will be kept.



3.2.1 Data Flow Diagram

3.3 Use case Diagram

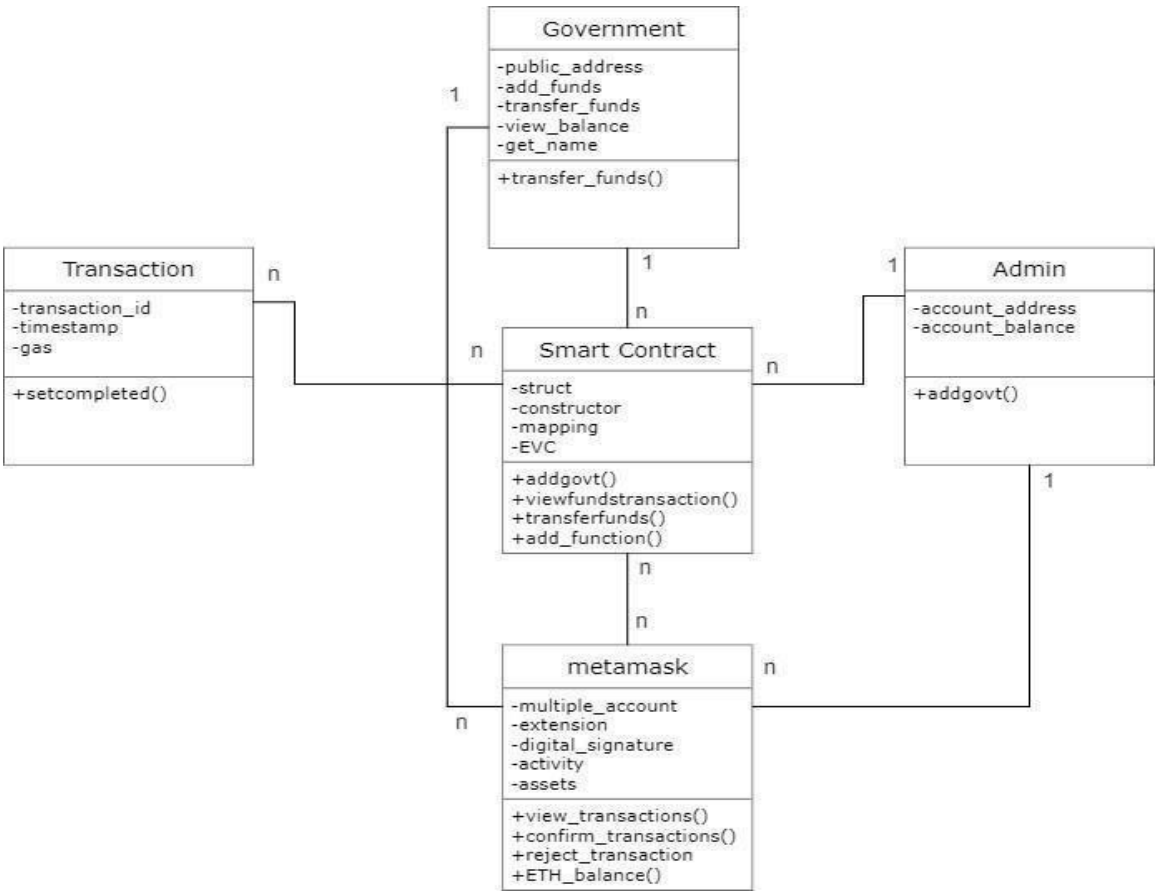
A Use Case is an interaction between the user and system. It ensures the goal of the users and responsibility of the system to its users. Admin, Government body and the guest are the three users interacting with the system as shown in Fig 3.3.1.



3.3.1 Use Case Diagram

3.4 Class Diagram

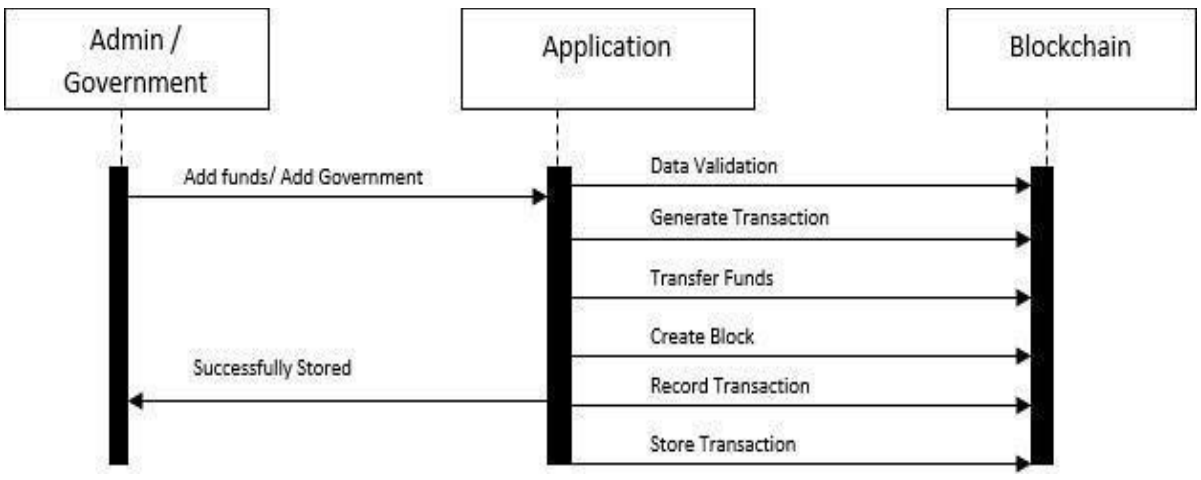
Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modelling of object-oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages. The diagram 3.4.1 shows the different classes and the attributes involved



3.4.1 Class Diagram

3.5 Sequence Diagram

A sequence diagram or system sequence diagram shows process interactions arranged in time sequence in the field of software engineering. It depicts the processes involved and the sequence of messages exchanged between the processes needed to carry out the functionality. The diagram 3.5.1 shows the functionality and different processes that take place in the application.



3.5.1 Sequence Diagram

Chapter 4

IMPLEMENTATION

Implementation basically deals with the implementing the design methodology into the state of working model. This is considered to be the very important stage of the project as this stage of the project aspects the system to the working procedure of the user. The developer implements all the design phase into the reality of the existing system to be used. Implementation is a process that puts the developed system into the actual use, along with the various working environment. The major task of the implementation stage is careful planning, looking through the system and various major constraint, the methodology design in such a way the changeover phase and also deals with evaluation.

4.1 Setting up Ganache

Ganache is a local development blockchain that can be used to mimic the behaviour of a public blockchain. It will allow us to deploy smart contracts, develop applications and run tests.

4.2 Installing node package manager

Once the local blockchain is running, we need to configure our environment for developing smart contract.

4.3 Setting up Truffle Framework

This provides a suite of tools for developing Ethereum smart contracts with Solidity programming language.

4.4 Setting up Metamask Ethereum Wallet

Most web browsers do not currently connect to blockchain networks, so we should install metamask to do so. Metamask will allow us to manage our personal account when we connect to the blockchain, as well as manage our Ether funds that we use to pay for transactions.

4.5 Deploy project

After making the project work on the local blockchain instance without any problem we can now able to deploy the project to the real test network and see how this application works in the real world.

```
function addGovernmentBody(address accNumber,string memory govtBodyName,string memory govtBodyRole) public {
    governmentBodyMapping[accNumber].accoutNumber =accNumber;
    governmentBodyMapping[accNumber].governmentBodyName = govtBodyName;
    governmentBodyMapping[accNumber].accountBalance=0;
    governmentBodyMapping[accNumber].governmentBodyRole = govtBodyRole;

    govtBodyNameMapping[govtBodyName] = accNumber;


    emit AddedGovtBody(govtBodyName,accNumber,govtBodyRole);

}
```

Figure 4.5.1: Implementation of smart contract (addGovermentBody)

```
function addFunds(uint funds,string memory time) public {

    require((keccak256(abi.encodePacked(governmentBodyMapping[msg.sender].governmentBodyRole))
    == keccak256(abi.encodePacked("Central"))));

    totalFunds+=funds;

    governmentBodyMapping[msg.sender].accountBalance+=funds;

    emit AddFund(funds,"Central Government",time);
}
```

Figure 4.5.2: Implementation of smart contract(addFunds)

```
function transferFunds(uint funds,string memory projectName,string memory govtBodyName,string memory timestamp) public {

    require(

    (keccak256(abi.encodePacked(govermentBodyMapping[msg.sender].govermentBodyRole))
    == keccak256(abi.encodePacked("Central"))) ||

    (keccak256(abi.encodePacked(govermentBodyMapping[msg.sender].govermentBodyRole))
    == keccak256(abi.encodePacked("State"))));

    address fundRecevier = govtBodyNameMapping[govtBodyName];

    require((keccak256(abi.encodePacked(msg.sender))) != (keccak256(abi.encodePacked(govtBodyNameMapping[govtBodyName]))));

    governmentBodyMapping[fundRecevier].accountBalance+=funds;

    governmentBodyMapping[msg.sender].accountBalance-=funds;

    emit TransferFund(funds,msg.sender,fundRecevier,governmentBodyMapping[msg.sender].governmentBodyName,governmentBodyMapping[fundRecevie

    emit YourBalance(governmentBodyMapping[msg.sender].accountBalance);

}
```

Figure 4.5.3: Implementation of smart contract(transferFunds)

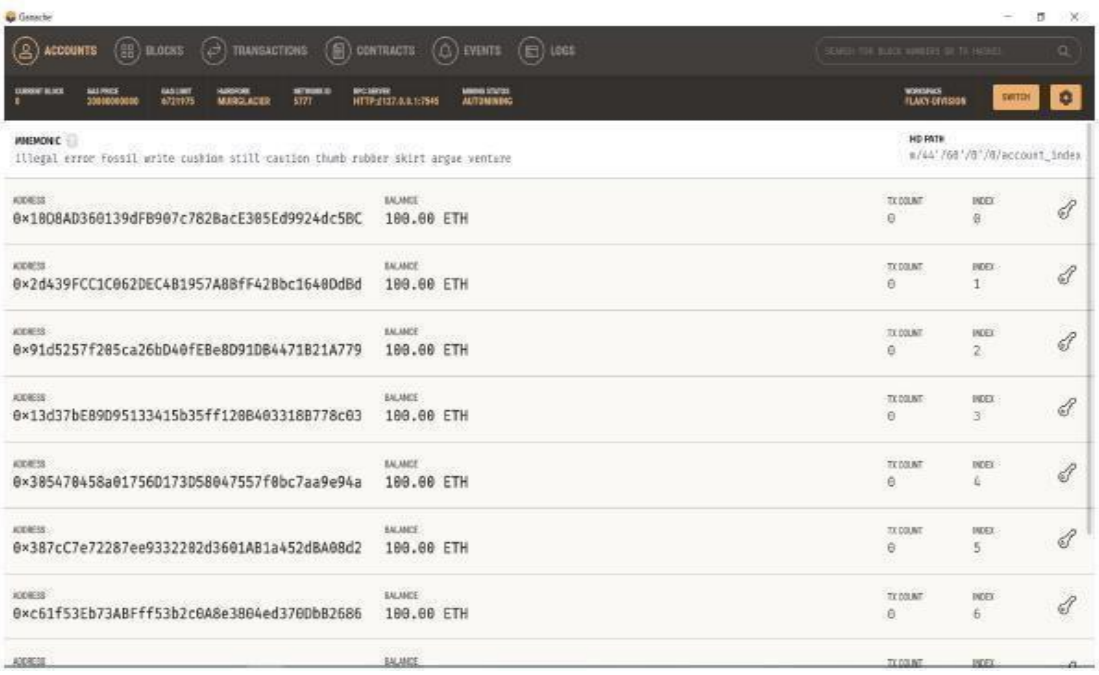


Figure 4.5.4: Ganache Home screen

The figure 4.5.4 shows the home screen of Ganache blockchain, with 10 user accounts with 100.00 ETH in each.

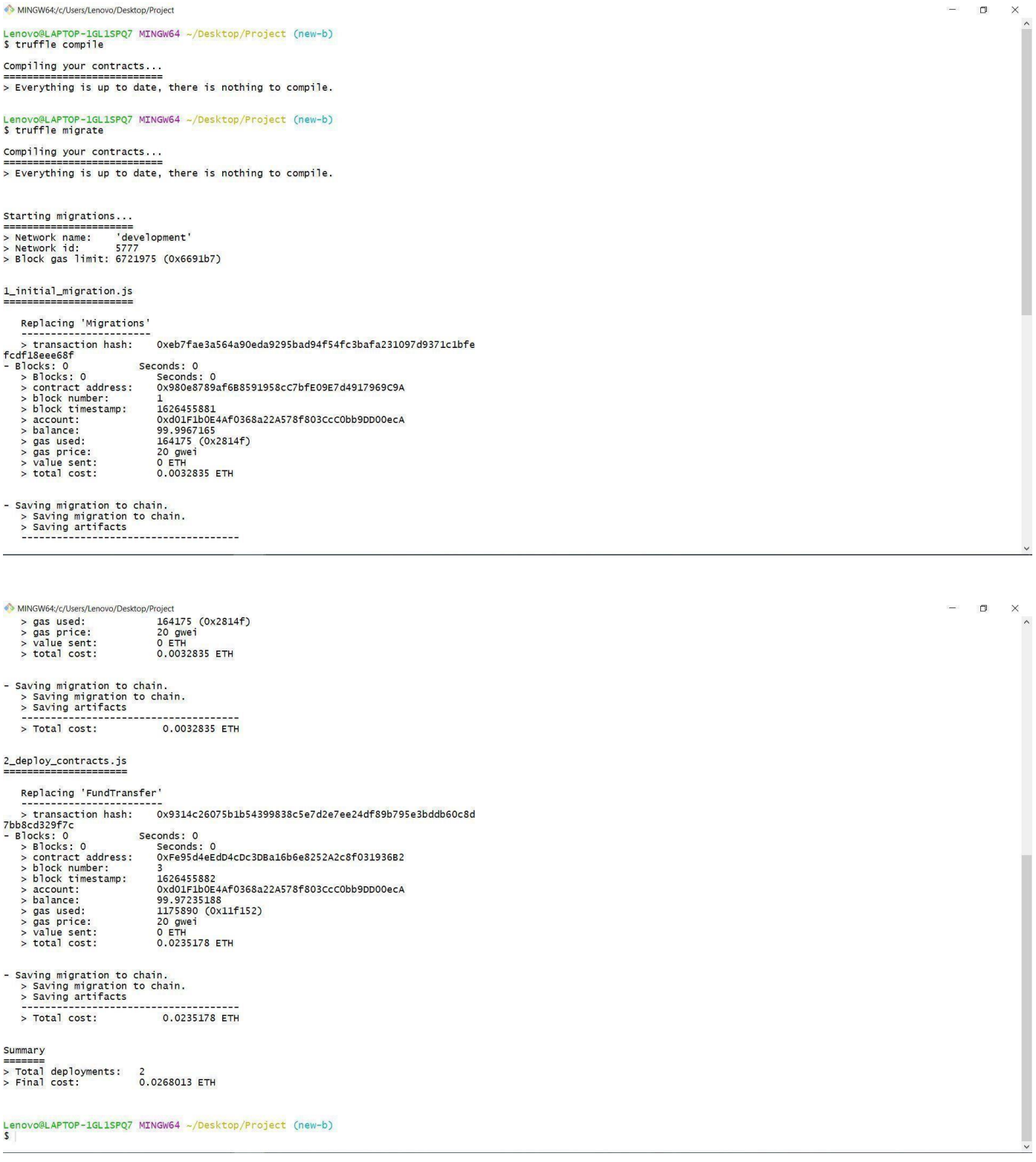


Figure 4.5.5: Truffle Migration

The Figure 4.5.5 shows the truffle migration using Git Bash. It takes few minutes to complete the migration.

```
MINGW64:/c/Users/Lenovo/Desktop/Project/client

Lenovo@LAPTOP-1GL1SPQ7 MINGW64 ~/Desktop/Project (new-b)
$ cd client

Lenovo@LAPTOP-1GL1SPQ7 MINGW64 ~/Desktop/Project/client (new-b)
$ npm run start

> client@0.1.0 start C:\Users\Lenovo\Desktop\Project\client
> react-scripts start

Something is already running on port 3000.

Lenovo@LAPTOP-1GL1SPQ7 MINGW64 ~/Desktop/Project/client (new-b)
$
```

Figure 4.5.6: Blockchain Network Deployment

Figure 4.5.6 shows the steps involved in deployment of blockchain network. Once the execution is completed the local blockchain network will be running at port 3000.

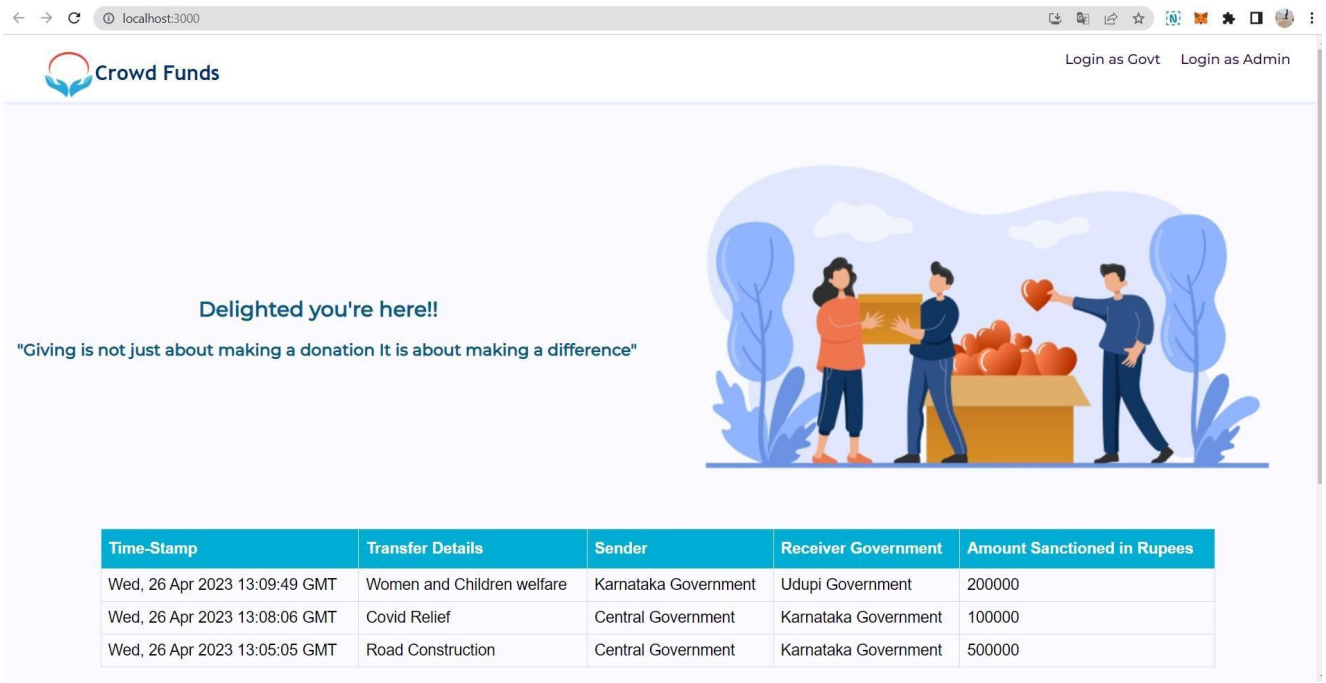


Figure 4.5.7: Home Page

The figure 4.5.7 shows the home page of the web interface used for government fund allocation and tracking.

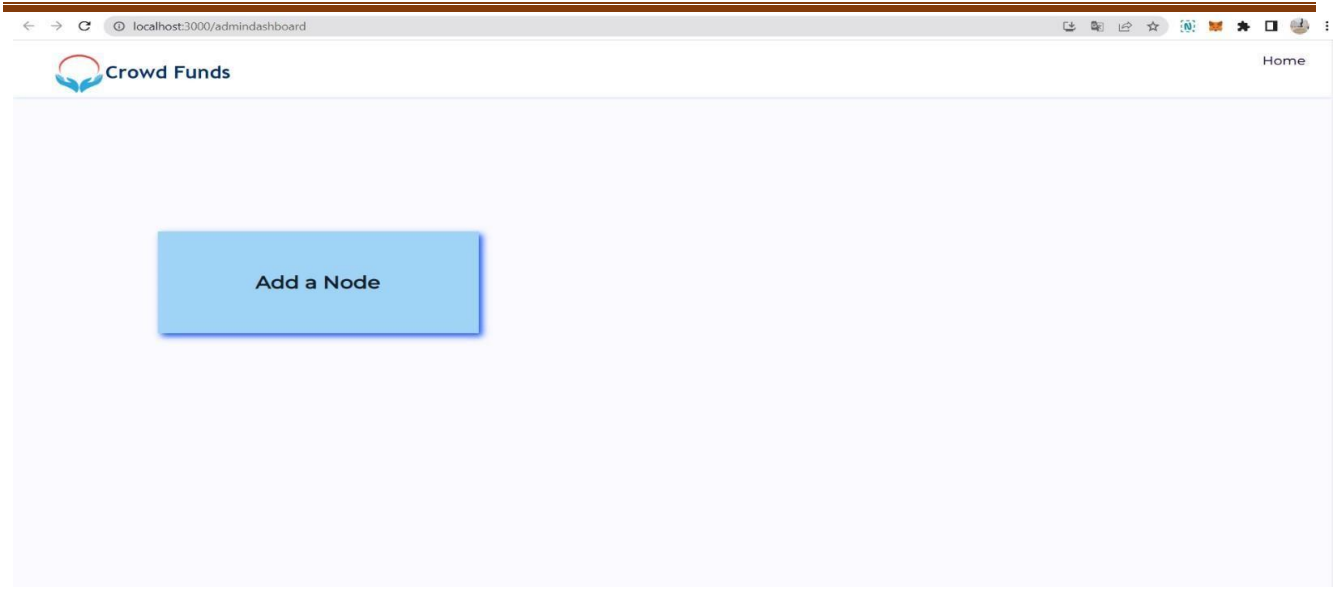


Figure 4.5.8: Admin Panel

The figure 4.5.8 shows the admin panel which consists of an option to add users to blockchain network.

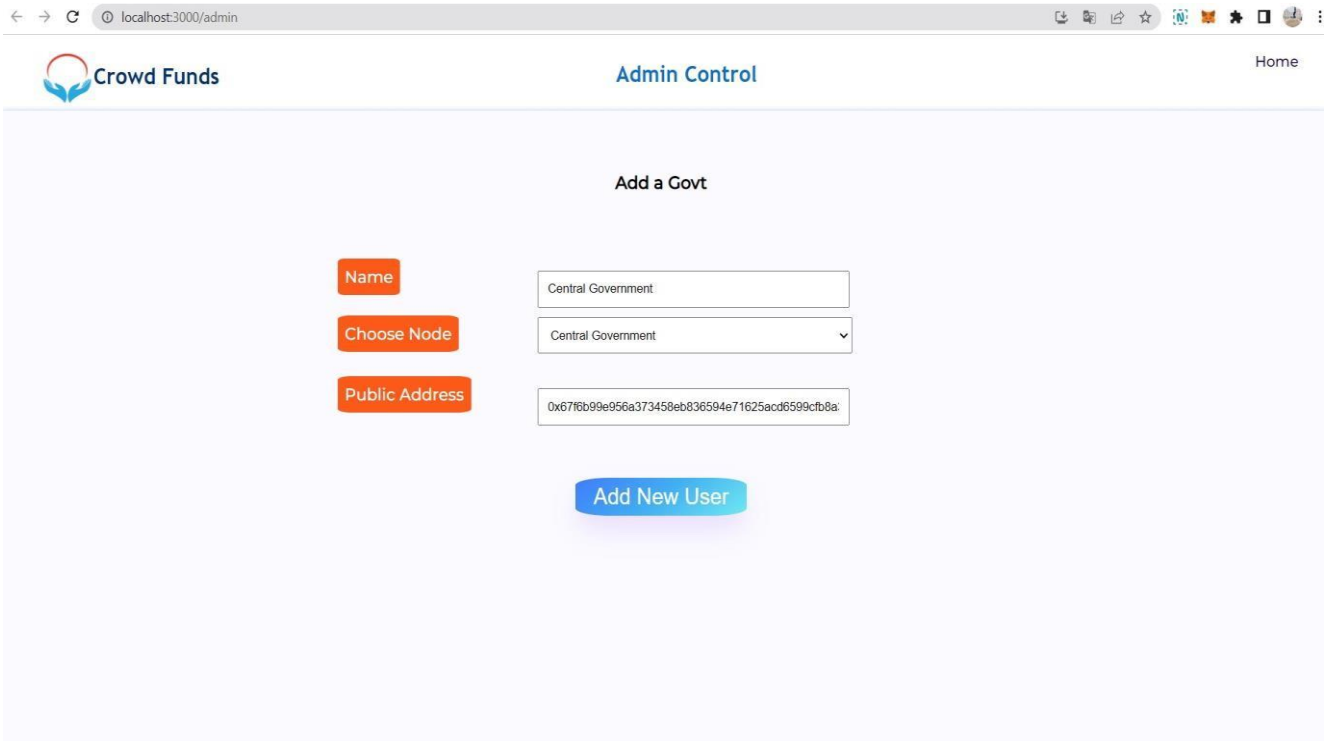


Figure 4.5.9: Adding Government bodies

The figure 4.5.9 shows the step of adding government body as user by admin.

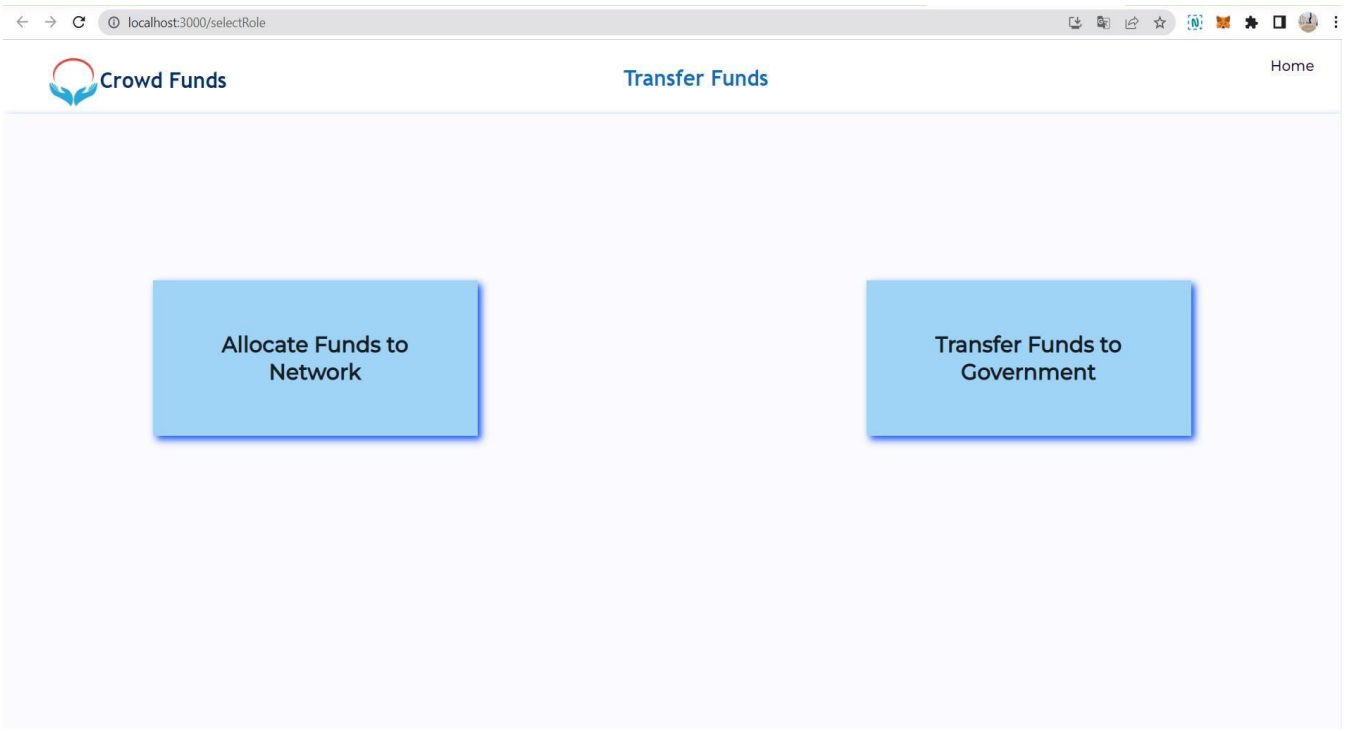


Figure 4.5.10: Login Options

The figure 4.5.10 shows the type of login the user can have, either to add fund or to sanction/transfer funds.

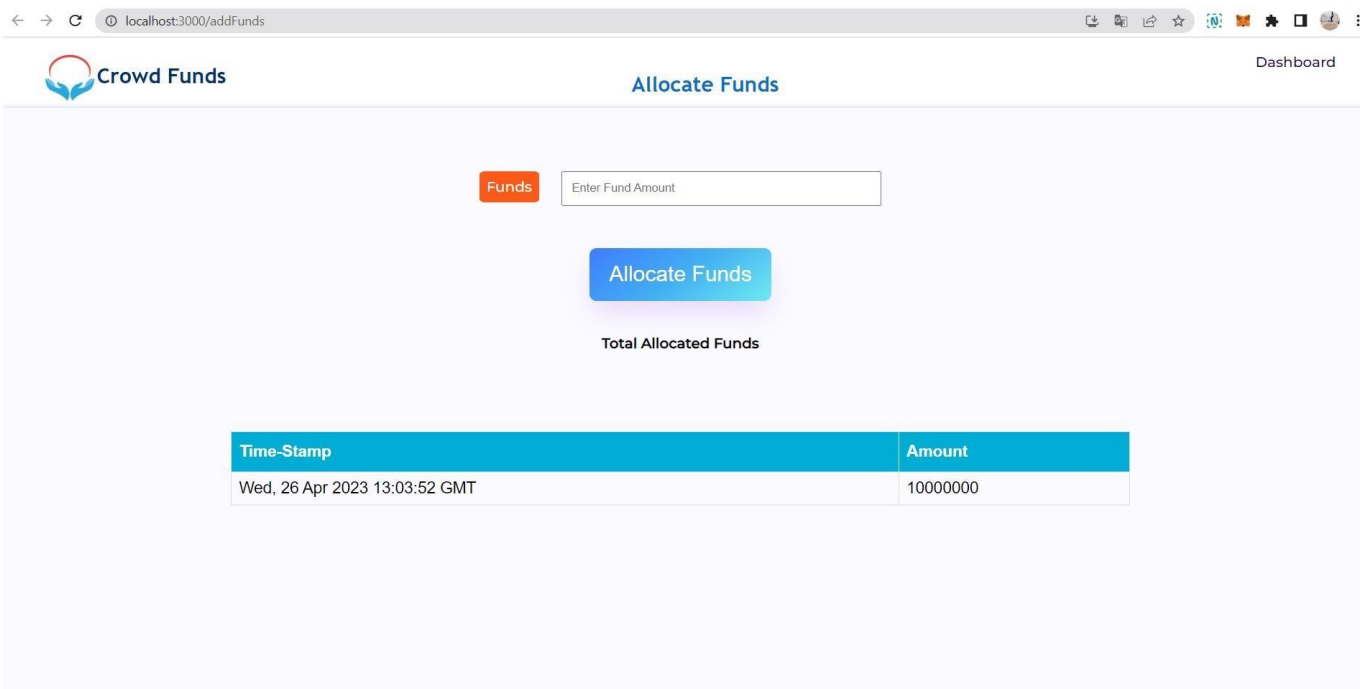


Figure 4.5.11: Fund Allocation

The figure 4.5.11 shows the way the fund is allocated or added to financial system of government.

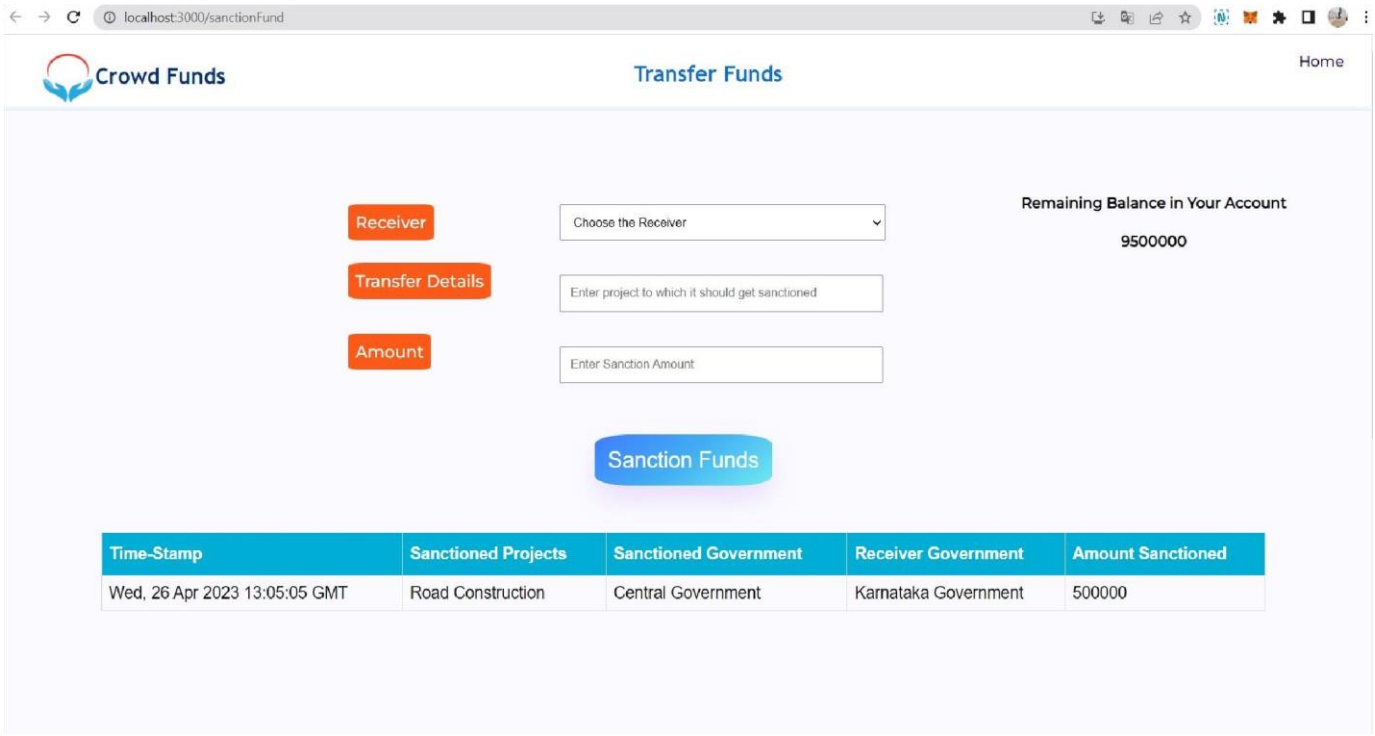


Figure 4.5.12: Fund Transactions

The figure 4.5.12 shows the transaction of funds across different government bodies.

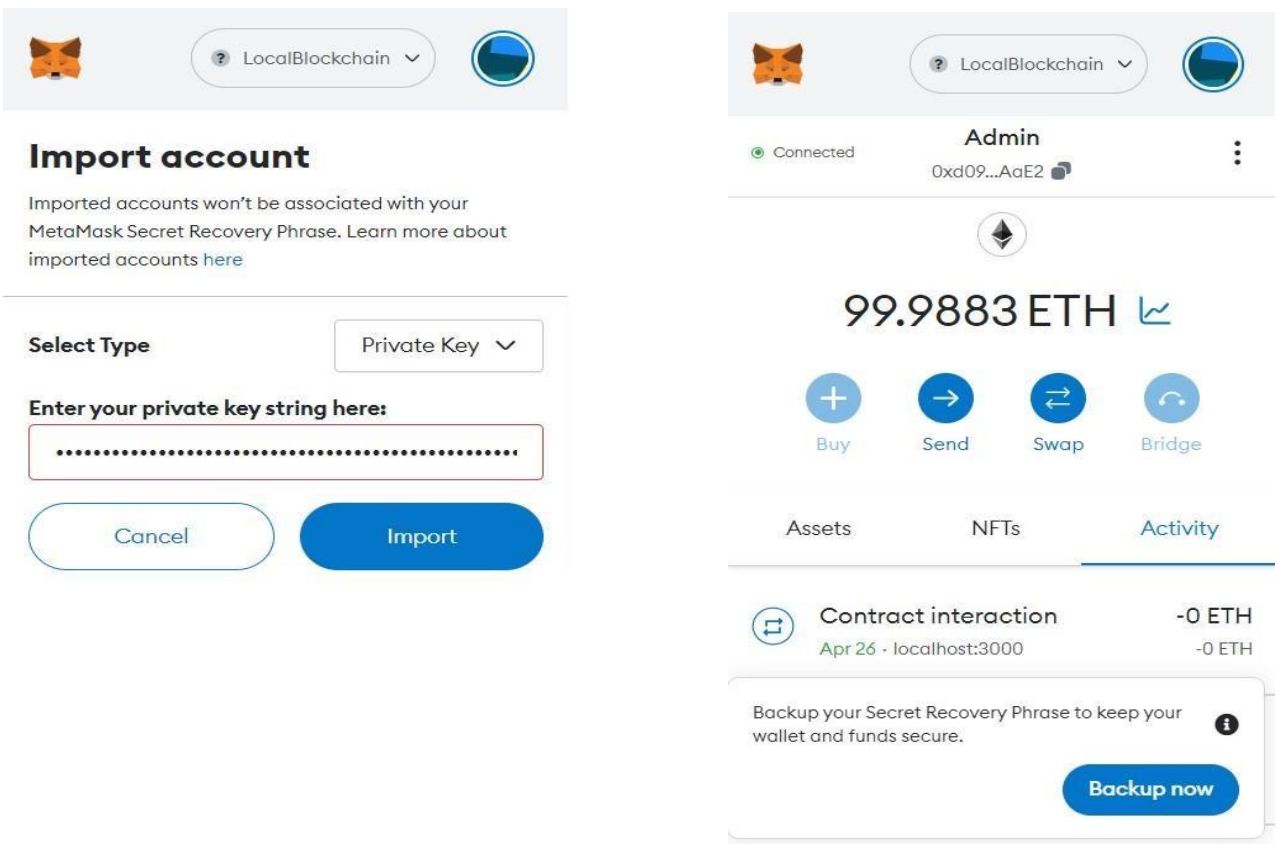
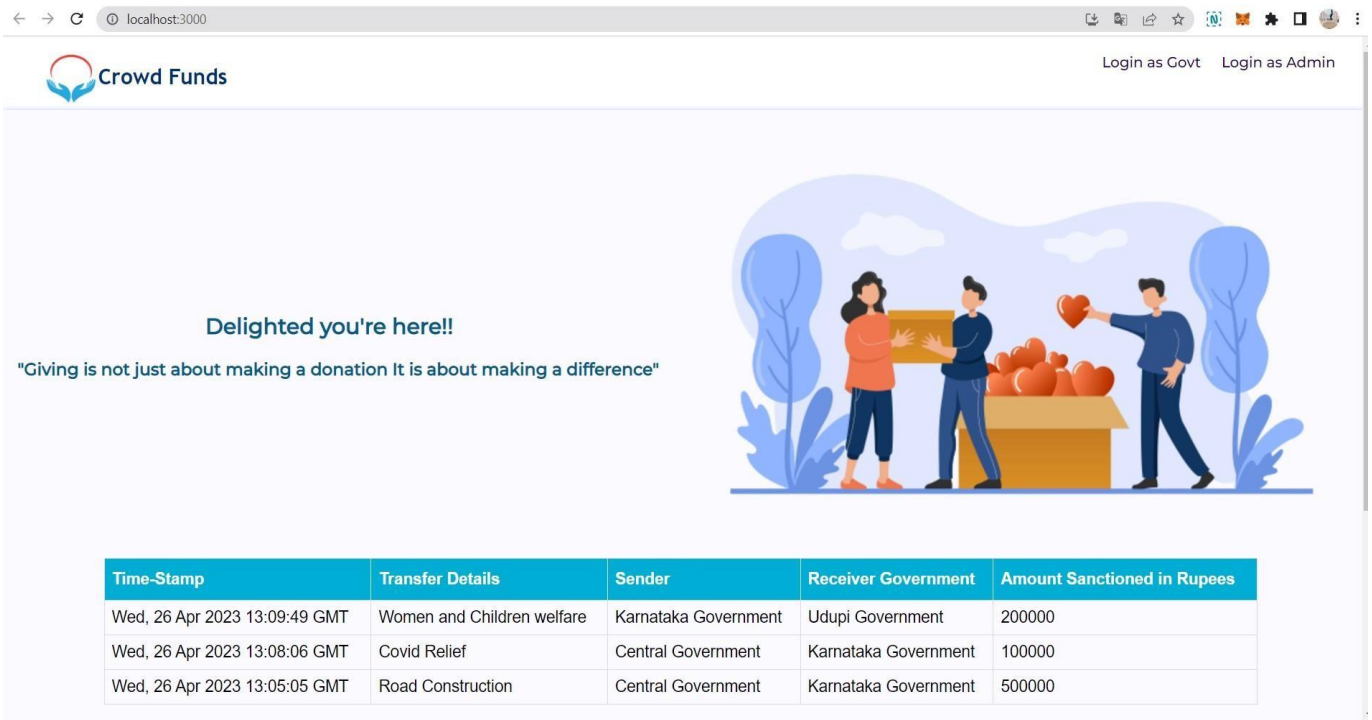


Figure 4.5.13: User login using Metamask

The figure 4.5.13 shows user login to metamask using the private key of the user

.



Time-Stamp	Transfer Details	Sender	Receiver Government	Amount Sanctioned in Rupees
Wed, 26 Apr 2023 13:09:49 GMT	Women and Children welfare	Karnataka Government	Udupi Government	200000
Wed, 26 Apr 2023 13:08:06 GMT	Covid Relief	Central Government	Karnataka Government	100000
Wed, 26 Apr 2023 13:05:05 GMT	Road Construction	Central Government	Karnataka Government	500000

Figure 4.5.14: Home Page Transaction Details

The figure 4.5.14 shows transaction history at homepage to all users including guest users.

Chapter 5

TESTING

Software testing is the process of investigating, verifying and validating software or applications to ensure that they are bug free and providing the stakeholders with information related to the quality of the software or service being tested. It gives the business a wider perspective to appreciate and understand the risks involved in software implementation.

Software testing involves two steps, mainly verification and validation of properties. In general, these properties indicate the limit to which the system under test meets the requirements that lead its design and development, if the system responds accurately to vivid inputs, performs its functions within a justifiable time limit, is easily usable, can be installed and run-in intended environments, and on the whole, if it achieves the expected result its stakeholders desire.

Software testing typically aims at executing a program or application with the intention of finding software bugs and also working on improving accuracy and efficiency. Testing is carried out in iterations and is an iterative process because when one bug is resolved, it may expose another.

5.1 Software Testing Levels

5.1.1 Unit Testing

In Unit testing, individual units of a software are tested. Unit testing is carried out to validate each unit of the software and its performance. This type of testing is usually done by developers on the go, to ensure that each unit is working and functioning as anticipated.

5.1.2 Integration Testing

Integration testing is where individual units are merged and tested together as a group. The aim of this level of testing is to expose incompatibilities, faults and irregularities in the interaction between integrated modules. Integration tests involve lot of code, and may produce traces larger than unit tests

.

5.1.3 System Testing

System testing is the process of testing the entire integrated system as a whole. This test is carried out to evaluate the system's end-to end compliance with the stated requirements.

5.1.4 Acceptance Testing

Acceptance testing involves testing a software product for acceptability. It is carried out to evaluate the system's conformance with the clients or stakeholders demands and assess whether it is acceptable for deployment.

5.2 Test Cases

TC #	Description	Expected Result	Status
TC - 1	Log into Metamask	Successful	Pass
TC - 2	Add Government	Added	Pass
TC - 3	Allocate Funds	Allocated	Pass
TC - 4	Transfer Funds	Transferred	Pass
TC - 5	Create block for each transaction	Block Created	Pass
TC - 6	Add block to Blockchain	Block Added	Pass

Table 5.2.1: Test Cases

Chapter 6

CONCLUSION

Blockchain decreases conflict and increases security by making it impossible to alter every block providing integrity, transparency. Cyber-Trust platform relies on advanced intrusion detection tools to identify malicious activities and enhance security.

The fund transfer information is safely stored as raw data in a blockchain database, while hashes and meta data of the transactions are stored on the blockchain.

Chapter 7

REFERENCES

[1] A. Mohite and A. Acharya, “Blockchain for government fund tracking using hyper-ledger,” in International Conference on Computational Techniques, Electronics and Mechanical Systems (CTEMS), 21-22 Dec., 2018.

[2] T. A. Syed, A. Alzahrani, S. Jan, M. S. Siddiqui, A. Nadeem, and T. Alghamdi, “A comparative analysis of blockchain architecture and its applications: Problems and recommendations,” 04 December 2019.

[3] Blockchain challenges and opportunities: A survey, booktitle=International Journal of Web and Grid Services, Volume 14, No. 4, author=Zibin Zheng and Shaoan Xie and Hong- Ning Dai and Xiangping Chen and Huaimin Wang, year=2018,"

[4] R. Chatterjee and R. Chatterjee, “An overview of emerging technology: Blockchain,” in 3rd International Conference on Computational Intelligence and Networks (CINE),28 Oct., 2017.

[5] H. R. Andrian, N. B. Kurniawan, and S. bah, “Blockchain technology and implementation:
A systematic literature review,” in 2018 International Conference on Information Technology Systems and Innovation (ICITSI),22-26 Oct, 2018.

[6] Hye-Young Paik 1, XIWEI XU 1, H. M. N. D. B. 1, S. U. L. 3, and Sin Kuang Lo 1, “Application of data management in blockchain-based systems: From architecture to governance,” in IEEE Access PP (99):1-1, Volume 7, 2019.

[7] M. Dabbagh, M. Sookhak, and N. S. Safa, “The evolution of blockchain: A bibliometric study,” 29 January 2019.

[8] S. Singh and N. Singh, “Blockchain: Future of Financial and cyber security," in 2nd International Conference on Contemporary Computing and Informatics (IC3I), 4-17 Dec. 2016.

[9] S. Tikhomirov, “Ethereum: state of knowledge and research perspectives,” in International Symposium on Foundations and Practice of Security, 206-221, 2017.

[10] S. Zhai, Y. Yang, J. Li, C. Qiu, and J. Zhao, “Research on the application of cryptography on blockchain,” in Journal of Physics: Conference Series, Volume 1168, Issue 3, 2019.