An Utility Bill and Government Toll Payment System using Blockchain Technology

By

Sajidul Islam
Roll: 1707010
&
Samiul Baree
Roll: 1707035

Khulna University of Engineering & Technology
Khulna 9203, Bangladesh
February 2023

An Utility Bill and Government Toll Payment System using Blockchain Technology

By

Sajidul Islam
Roll: 1707010
&
Samiul Baree
Roll: 1707035

A thesis submitted in partial fulfillment of the requirements for the degree of
Bachelor of Science in Computer Science and Engineering

Supervisor:

Dr. M.M.A. Hashem
Professor
Khulna University of Engineering & Technology
Khulna, Bangladesh

Signature

Khulna University of Engineering & Technology
Khulna 9203, Bangladesh
February 2023

Acknowledgements

First of all, all the praise goes to the almighty Allah, who helped us with all His blessings and kindness for us to be able to finally see our thesis work come to fruition. It is our honour to be guided in the process by our supervisor Dr. M.M.A. Hashem, Professor of the Department of Computer Science and Engineering (CSE), Khulna University of Engineering Technology (KUET). We will ever be grateful to him for constantly remaining supportive and encouraging throughout the process despite our own shortcomings. We are thankful to our families, friends, and all people connected to our lives who played any kind of role in making this work possible. Finally, we would also like to express our gratitude to our friends for supporting us by lending a hand as we worked on our thesis. Last but not least, we would want to express our gratitude to all of our deserving professors, staff members, department authorities, and friends and family. They have always been keen to work together to ensure the success of this research.

Authors

Abstract

Blockchain technology is currently a highly hot topic in the security business when it comes to secure transactions or currencies. The security of transactions was strengthened by creating cryptocurrency. But blockchain technology has applications beyond just cryptocurrency. It can guarantee decentralization such that a system doesn't need a centralized medium to run or maintain it; rather, peers or users are in charge and they are the ones who make the system as a whole work. One use case for these advantages of blockchain is the payment of taxes or utility bills. The use of utilities like electricity, water, and gas requires that we pay a certain portion of the bill. Although implementing blockchain technology can assist implement these features while also providing maximum security and simplicity and preventing tampering with data, transactions, bill amount, etc., the process of paying bills can be secure. As was already said, blockchain is a decentralized secure background system. Our suggested approach eliminates the central medium that was one of the causes of data tampering and ensures the security of billpayer data, transaction histories, etc. These also ensure the long-needed transparency that has been missing from the current systems.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Blockchain is a distributed ledger where blocks are connected to form an immutable chain. By changing just one element in a block, you can change the identity of the entire block and the chain preceding it. Due to the decentralized nature of the system, transparency between peers is guaranteed because each peer has a copy of the same blockchain. As every peer has the same blockchain, tampering with a specific peer's chain results in all the other peers' blockchains detecting the temperament and fixing it by restructuring the blockchain for that particular peer. This makes it difficult to manipulate blockchains because changing a single element in a block changes the whole block identity and prevents that block from existing in the chain as the previous or next blocks cannot identify the block.

The concept first appeared in a paper titled A Peer-to-Peer Electronic Cash System [3], which was published in 2009 by Satoshi Nakamoto, an individual or organization. which eventually results in the introduction of bitcoin, the first blockchain application ever. According to the paper, the sender should sign each transaction with a hash to aid in monitoring subsequent transactions that create a chain.

A white paper on Ethereum, a different cryptocurrency from bitcoin that is more flexible than bitcoin because it supports a much wider range of applications using the blockchain concept than just using it for online transactions, was published by Vitalik Buterin sometime after the release of bitcoin. It introduces ideas like smart contracts and gives birth to the first blockchain approach based on applications.

The government of Bangladesh has a number of agencies that provide the citizens of the nation with various utility services, including power, gas, water, and the internet. People pay these service providers a set sum based on how much they use their services. These organizations include BTCL, DPDC, DWASA, several road toll systems and others. Equations

are used to determine the cost of these services based on how much each user uses them. In the past, customers had to physically visit the offices of the relevant service providers or the bank to pay their expenses. These payments can now simply be made online using a computer, a smartphone, etc. To make the payment process online and accessible to every user in their house, providers developed their own systems while collaborating with different third-party media. So, the hassle of going to places physically got removed by this step of digitization and made easier for the users. Therefore, this phase of digitization removed the inconvenience of physically visiting sites and made it simpler for users.

1.2 Motivation

Bangladesh is a third-world nation with corruption present in every conceivable industry. It has numerous financial problems, including fraud, black money, and inflation. Even though the nation is heading toward digitization, fraud is still a possibility on every level. These days, we pay our monthly bills for utilities like electricity, water, and gas using a variety of online methods like Bkash, Nagad, and iPay. They charge a fee for using their services, which might occasionally be expensive. Additionally, the public is not always aware of the procedure used to transfer money to the government and how it is correct. This creates a space for resource abuse and information manipulation. If we want to actually go toward transparent management, this level of transparency must be assured. The benefits of blockchain technology include procedural transparency, the near-impossibility of data tampering, and the ability for users to send money directly to the government or any other institution they are paying for. So, the problems with the bill payment procedure can be resolved with the help of the blockchain technique.

1.3 Problem Statement

In Bangladesh, payment systems for utilities, VAT, and tolls are now available online. But they are not always secure and are vulnerable to hacking. Due to their high vulnerability to cyberattacks and the fact that Bangladesh is a third-world country, other countries' hackers

may find it simple to target Bangladesh. Additionally, dishonest individuals like tax fraudsters may falsify data or alter the system for their own self-serving purposes, putting regular users or taxpayers in a position where their hard-earned money might well be stolen without anybody noticing. The value of foreign money is rising steadily, and there is little evidence of its stability at the moment. This adds to the problem of inflation in Bangladesh. Compared to other countries, the value of the Bangladeshi currency is strangely falling. Since cryptocurrencies like Ethereum, which are prone to inflation and have their ups and downs but whose economic rate graph is always exponential and rising over time, may be used in place of traditional payment methods for taxes or bills, this can greatly improve the situation.

1.3.1 Recent Flaws in Toll Payment System

1. **Extortion:** There are numerous claims made by automobile owners and drivers that they are subjected to a lot of illegal collections from both the police and those responsible for this toll-collecting system. Additionally, local thugs steal money from toll-free businesses as well. Most of these horrible atrocities happened at night.

2. **Time Consuming:** Toll collecting by hand takes a lot of time. When there is a lot of traffic, large traffic congestion (a few miles long) happens, which has very serious consequences such as raising the cost of goods, making it difficult for sick people to get about, and delaying the delivery of rotting goods, among other things. The economy of a nation would suffer greatly as a result of this time consumption. Additionally, if the vehicle doesn't have the precise retail toll amount in paper money, paying the toll can take a long time. This procedure could take a very long time.

3. **Need More Manual Labor:** A lot of people are needed for the entire toll collection procedure, including toll collectors, security personnel, financial experts, etc. The traditional toll payment mechanism is still being questioned even after this significant manpower effort.

4. **Toll Fraud:** In addition to abusing their authority to use these public facilities without paying tolls or taxes, some people also abuse their authority to collect extra money from automobiles. But everyone should be subject to the same laws.

1.4 Proposed System

In this thesis, we suggest a mechanism for paying utility bills including electricity, gas, and water utilizing blockchain technology. To utilize the service, users will need to sign up and log in, and this user data will be recorded in the blockchain, making it difficult to tamper with that as well. No additional outside assistance will be required. We are using the Ethereum blockchain and the cryptocurrency ether (eth) since it allows us the flexibility of building smart contract scripts. Each user has a unique profile and bills to pay, thus it is necessary to create smart contracts to automate the transaction process and manage the bill systems for each user. Each user will transfer money from their individual cryptocurrency account to the given address. Every level of user will find the proposed system simpler to operate.

Since we are using smart contracts, they will control the transactions that are sent through the blockchain, signing them with the user's private key (each user will have a private key connected with their crypto account), and then adding them to the blockchain after they have been encrypted. Every transaction is uploaded to the blockchain inside of a single block, giving every transaction a timestamp when it is distributed.

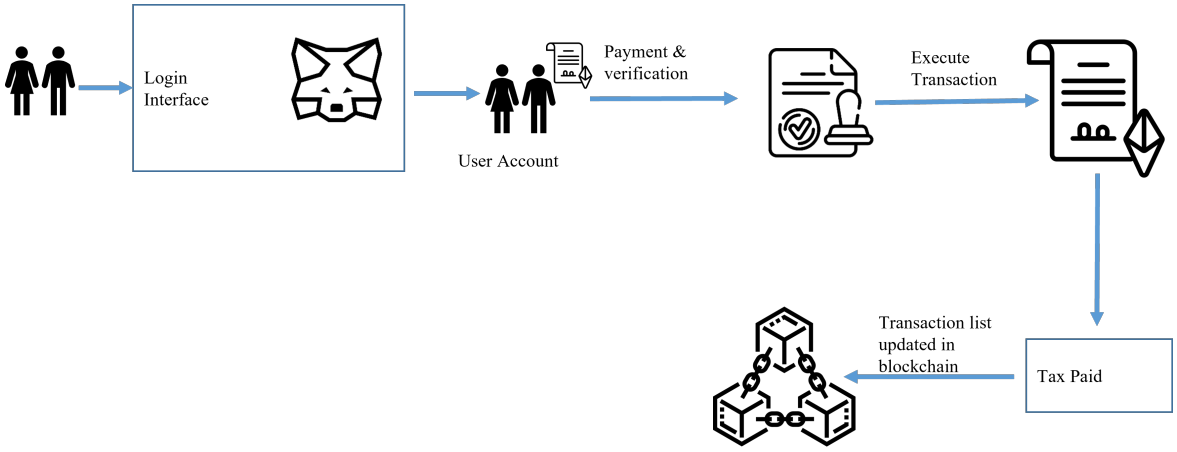

FIGURE 1.1: Proposed Model

Users choose the service they wish to pay for, and an account address for that service is automatically chosen. Users then choose an account that has enough Ethereum to cover the

payment, and they can confirm the transaction by signing it. A smart contract then deploys the transaction to the blockchain, and the recipient account receives the users' payment.

1.5 Contribution

This thesis study has primarily contributed to the development of a contextually appropriate technique for our national perspective. For adaption to the Bangladeshi legal framework, we presented a blockchain-based utility bill and government toll payment system.

In order to complete this thesis, we used the Ethereum platform to prototype a working version of our suggested paradigm. We also develop a suitable smart contract that can facilitate the bill payment system's operation, as smart contracts are a key component of Ethereum-based blockchain applications. To make it simple for the user to use the services simply, we created a user interface (UI). Our user interface includes the typical login and registration sections where users enter their information. However, what sets our authentication apart from other authentication methods is that we store user data on a more secure blockchain by spending a little amount of gas. Users can link their cryptocurrency wallets if they have enough Ethereum in their accounts to cover the fee. They can choose the exact services, such as electricity or gas, for which they want to pay their bills from a variety of providers, and the appropriate account address will be selected automatically. The user then chooses the account from which they wish to pay the bill, confirms the transaction, and signs it. Smart contract then will deploy the transaction to the blockchain in a block and thus the process will be completed. The procedure is then finished when the smart contract deploys the transaction to the blockchain in a block.

Chapter 2

Literature Review

There are several research studies regarding the use of blockchain and blockchain-based smart contracts. As a potential technology to mitigate fraud in bill and toll payment systems. After the initialization of Ethereum in the last decade there is a significant development in smart contracts, solidity, and decentralized applications using Ethereum blockchain.

There is a study in [4] that shows that blockchain has a very important effect on tax systems around the world. Government facilitates in many ways using blockchain-based tax payment systems. Some country is trying and some already integrated their tax and other administration services with blockchain. Integrating blockchain in public administration helps increase the security of important infrastructure and data, facilitate transparency, etc. [5]. Making decentralized applications for the purposes of integrating blockchain and regular important but vulnerable systems is a very effective approach. A decentralized application is an application that is run by multiple users over a distributed network run by the computational power of every user [6]. The author of [6] also stated that after researching thousands of dapp usage the growth of dapp usage is exponentially rising and the active users in dapps are rising as well. To create decentralized applications (dapp) we need to write dapp to make it based on blockchain.

Another study is presented about a blockchain-based smart contract prototype [7] for a specific benefit process from the Syddjurs Municipality government in Denmark about the deployment of blockchain-based smart contracts for municipal government processes. The authors show that there are some benefits to adopting that technology for the government processes such as integrity guarantees, verifiability, and direct collaboration of payment between parties. There are also some problems detected such as the cost of latency, peer-to-peer transaction charges, the immutability of errors, and a very concerning single point of failure the municipal government which is losing blockchain private keys resulting in losing control over government casework, with no resources.

According to a study in [8] Blockchain technology in the VAT system has the potential to strengthen the system and give more trust to the parties in the network, including the government and TE. With a system that provides data openness, blockchain technology is an open technology innovation in the VAT system and particularly in e-invoicing, because it can increase the system transparency and efficiency. Transactions in blockchain can be tracked in a completely decentralized environment [9] so transparency and indisputability features will be there. It can also reduce discrepancies and save time in the payroll industry of the government.

In Saudi Arabia, [10] gives a system for the country's newly added Value Added Tax (VAT). In the system, peers will have their own stakeholders which include various supply chains like manufacturers, mediocre, etc. The process is troublesome for governments, if certainly feasible, to track VAT installments. The computerized age is additionally forming tax collection systems into a totally extraordinary shape, by not just changing the connection between citizens and tax authorities, yet in addition, modifying the manner in which government taxes are covered, submitted, and stored. The transparency, security, immutability, and cost-effectiveness of the process make it more viable than any kind of regular approach.

Chapter 3

Related Technologies

3.1 Blockchain

Blockchain technology has started to define and shape new areas of information technology and computer science. The necessity for a decentralized currency was previously more of a theoretical idea, but in the last ten years, it has become a reality owing to Satoshi Nakamoto's well-known article from 2008, which introduced Bitcoin and blockchain technology [11]. A Blockchain transaction can be defined as a small unit of a task that is stored in public records. These records are also known as blocks [12]. A block is declared by a collection of transactions, the previous block's hash, and a nonce. A timestamp server hashes a block and publishes the result, demonstrating that the information inside the block must have been there at the time of hashing. The timestamp server must confirm that the block's timestamp is less than two hours in the future and higher than the timestamp of the chain's previous block. These blocks are executed, implemented, and stored in the blockchain for validation by all miners involved in the blockchain network. Each previous transaction can be reviewed at any time but cannot be updated [13].

As seen in Fig. 3.1, a blockchain is created by connecting these hashes in a chain [11]. Every block contains the hash of its predecessors, which validates the relationship between two particular blocks and creates a blockchain. Using a straightforward block change that blocks the hash and, as a result, does not match the hash saved in the following block, the blockchain rejects that order and refuses to accept the change.

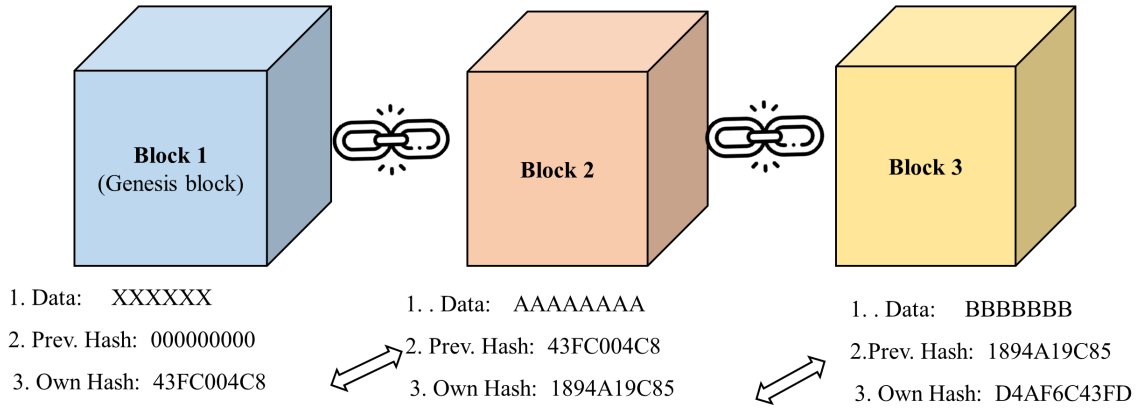

FIGURE 3.1: The Blockchain Scheme

Every block of a blockchain consists of various transactions Each transaction is kept in a pool of unconfirmed transactions, and the Gossip protocol, a flooding mechanism, is used to spread it throughout the network. Peers then need to select and validate these transactions based on a set of predefined criteria. For instance, the nodes attempt to authenticate and verify these transactions by determining whether the initiator has enough balance to start a transaction or by causing double-spending in an effort to trick the system [12]. Using the same input amount for two or more distinct transactions is known as double spending [13]. The transaction is included to a block once it has been examined and approved by the miners. Miners are peers who mine for blocks using their computational power [14]. A digital distributed ledger and a digital timestamp serve as the record and verification, respectively, of every transaction that takes place within a blockchain network. As a result, by gaining access to any network node, it is feasible to audit and trace earlier records [15]. For instance, Bitcoin allows for the iterative tracing of all transactions, facilitating the auditability and transparency of the data state in the blockchain. However, it becomes exceedingly challenging to track down the source of the money when it is spread across numerous accounts.

Blockchain technology is used to record transactions across numerous computers in a way that cannot be changed or destroyed. It is a decentralized ledger of transactions. It promotes accountability and confidence without the need for a centralized authority because it is secure and transparent. The consensus techniques, like as proof-of-work and proof-of-stake, among others, make it challenging to modify a blockchain. A new block of transactions can be uploaded to the blockchain after all nodes or computers have contributed to reaching a

consensus on that block. Because there is no need for a central authority and because every peer or user in a chain has access to the same information or copy of the chain, the system is also made decentralized.

By implementing these capabilities, Ethereum's blockchain has significantly increased the versatility of using it for solving problems in everyday life as well as cryptocurrencies like Bitcoin and Ethereum. The creator of the Ethereum platform, Vitalik Buterin, introduces the terms ”smart contract” and ”decentralized apps” in the Ethereum white paper [12].

3.2 Ethereum

Vitalik Buterin, the co-founder of Bitcoin Magazine, introduced Ethereum in 2014. Following a year of successful use of bitcoin, it was discovered that implementing a straightforward bitcoin approach in other applications is challenging because bitcoin itself has various restrictions. For instance, Bitcoin has scripting restrictions such as Turing completeness, UTXO cannot allow for control over withdrawal amounts, a condition known as value-blindness, UTXO can only ever be spent or unspent, and there can never be a multi-stage open contract. Additionally, UTXO lacks the value of the nonce and previous hash, which severely restricts the application [12].

As a result, Ethereum has been created as a brand-new blockchain with the benefits of an integrated Turing-complete programming language that enables anybody to create smart contracts for decentralized applications where they may establish their own rules for ownership, transaction forms, etc. The ”accounts” that makeup Ethereum are objects with four fields. 1) A counter called once is used to ensure that each transaction is handled just once, 2) Current balance of the account, 3) Account contract code, 4) Account storage (which is empty by default) [3].

Ethereum is the smartphone of blockchains: a platform that allows for the creation of ”apps” that can be used immediately by Ethereum users without the need to download any additional software [16]. The pattern is a little more intricate in Ethereum. The collection of all accounts, where each account is either a contract or an externally owned account (EOA), can

be referred to as the state. If the account is an EOA, the state merely saves the balance of the account in ether (Ethereum's internal crypto-token, comparable to bitcoin or XRP in function) and a sequence number intended to thwart transaction replay attempts. If the account is a contract, the state keeps a key-value database with the contract's code and storage information. On July 30, 2015, the Ethereum blockchain officially went live. A proof-of-work (PoW) protocol was used to secure the chain for more than seven years. As the network difficulty rose, the miners' energy requirements climbed. Using de Vries's technique, the network's annualized electricity usage peaked on August 13, 2022, at 93.975 Terra Watt Hours. For context, the Philippines use more than this.

The network's creator, Vitalik Buterin, had called for a switch to proof-of-stake (PoS) as early as 2016. Stakeholders are required to confirm new transactions using this validation method. Instead of looking for the nonce at random, stakers use their Ethereum holdings as collateral in a smart contract. Staker's staked coins may be forfeited if they unintentionally or maliciously fail to carry out their validation obligations. The RANDAO pseudo-random algorithm is used to select the scorers for any given block. Since proof-of-stake (PoS) avoids the energy-intensive problem-solving feature of proof-of-work, the Ethereum network's electricity consumption has dropped to 0.015 Terra Watts, a 99.98% reduction, as a result of the switch to PoS.

3.3 Smart Contract

A blockchain transaction's automation is carried out through Smart Contracts, a Turing-complete programming language that is a part of Ethereum. Smart contracts, in their most basic sense, are agreements made directly between two parties. A "smart contract" is a computerized transaction protocol that carries out a contract's condition, according to Nick Szabo, who first used the term in 1994 [13].

A smart contract is a piece of code that executes on the Ethereum blockchain and sends transactions for various purposes. It is superior to traditional contracts because it operates on the blockchain, which eliminates the need for middlemen between parties to transactions. Additionally, fewer intentional or unintentional exceptions exist since contractual clauses (such as collateral and bonding) are translated into code and integrated into hardware and

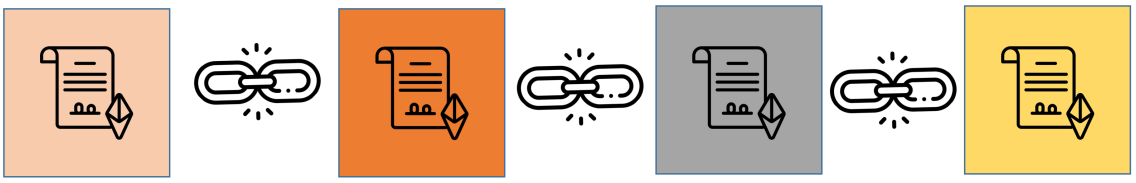

FIGURE 3.2: Smart Contract

software that may self-enforce them [14]. Smart Contracts can be on-chain or off-chain. On-chain smart contracts are generally executed in the blockchain with all relevant transactions being visible to the entire blockchain network. This reduces the privacy of smart contracts. User also has to pay a certain amount of gas fees to send transactions and transactions can take some time. As for Off-chain smart contracts they are executed outside of the blockchain. An off-chain smart contract only needs to be signed and executed by the interested participants. It is designed to execute high computational works for which online transactions can be costly [17].

On the Ethereum blockchain, we are running smart contracts for our system. Contract bytecode is executed using the Turing complete Ethereum Virtual Machine (EVM). Automation, self-sufficiency, and decentralization are properties of smart contracts. Decentralized autonomous organizations (DAO), decentralized autonomous corporations (DAC), and other types of autonomous organizations can be created using smart contracts thanks to their features or behaviors [18].

There are four sequential phases that make up the entire life cycle of smart contracts [19]. 1) Creation of a smart contract, 2) Deployment of a smart contract, 3) Execution of a smart contract, 4) Completion of the smart contract. As the smart contract is the digital and transparent version of the physical contract smart contract works like it. Starting with the fundamental distinction between an agreement and a ”contract,” a consideration of the enforceability of smart contracts is necessary. Although two parties can enter into a variety of ”agreements,” most states acknowledge that a contract is an agreement that is enforceable in a court of law and has legal effect. State courts typically assess whether the offer, acceptance, and consideration requirements under common law are met in order to determine enforceability. Ancillary smart contracts can definitely help to meet these fundamental needs. For instance, an insurer might create a product for flight insurance that automatically pays the insured if a flight is delayed by more than two hours. The essential terms, like defining how the delay is

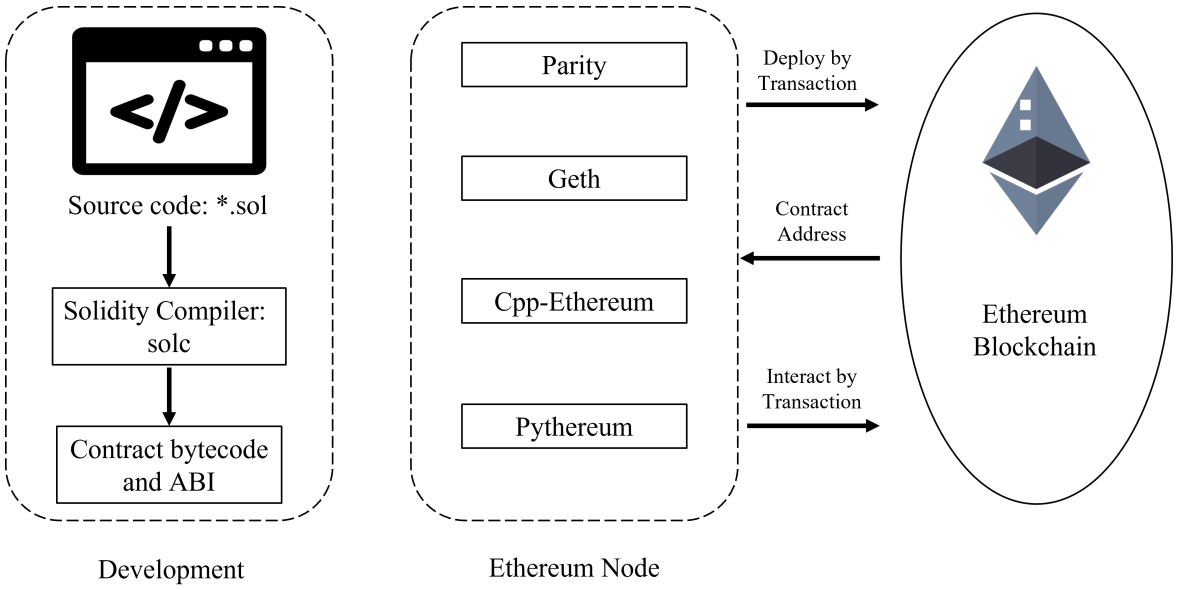

FIGURE 3.3: The process of smart contract development, deployment, and interaction

determined, can be stated in a text-based contract, with an auxiliary smart contract handling the formation of the contract (payment of the premium) and execution (automatic payout following a verifiable delay). In this case, the insurer has issued a firm offer for a flight insurance policy, which the insured accepts in exchange for paying the premium as payment in full.

Given the current legal frameworks for accepting electronic contracts, it is possible that a court would uphold the legality of code that carries out smart contract clauses, or what we have dubbed auxiliary smart contracts, today. Therefore, the obstacle to widespread smart contract adoption may have less to do with legal restrictions and more to do with potential conflicts between the operation of smart contract code and how parties conduct business.

3.4 Solidity

For the purpose of deploying smart contracts on the Ethereum blockchain, the programming language Solidity is employed. Solidity is built using the Ethereum Virtual Machine (EVM), which also creates the environment in which it can be operated. Programmers can create a variety of functions using Solidity to create smart contracts that are both cost-effective and

efficient. Solidity is still developing, growing, and becoming more adaptable and scalable every day. Now it offers a tool for making custom calls to utilize storage as efficiently as possible.

Modifiers are a very helpful aspect of Solidity. Modifiers are encapsulated code units that take arguments and affect the way the code is executed. This technique makes use of the condition-oriented programming (COP) paradigm to get rid of conditional routes in function bodies. Modifiers are used to quickly alter a function's behavior; they are specified after the function's name in a list after a space. Function modifiers are supplied after the function name in a list that is separated by whitespace and are used to swiftly alter the behavior of functions. Modifiers are frequently employed to perform conditional checks prior to the execution of functions.

Events are yet another practical and fascinating feature of Solidity. Events that are dispatched signals can be fired off by smart contracts. Applications and user interfaces can listen to blockchain events and react to them without spending a lot of money. For the Ethereum platform, Solidity was initially intended as a typed, JavaScript-like programming language. The language is divided into definitions of contracts, functions, and modifiers, as well as statements and expressions. Contract definitions are the building blocks of a Solidity source unit [20].
Solidity is developing quickly for a language that is still relatively new. It aspires for one breaking release per year and a regular (non-breaking) release per month.

3.5 Merkle Tree

Merkle tree is a binary tree type [2]. The tree is made up of a lot of nodes, with an enormous number of leaf nodes at its base. The underlying data, a group of intermediary nodes where each node is the hash of its two children, and finally a single root node acting as the tree's "top" are all included there. The hash of the tree's two offspring also makes up the top. A node can receive just the block header from one source and a tiny bit of the tree that is relevant to them from another source and still be confident that all of the data is genuine since the Merkle tree is designed to allow data in a block to be transferred piecemeal. The

reason this works is that hashes spread upward: if a malicious user tries to insert a fake transaction at the bottom of a Merkle tree, the change will result in a change in the node above it, and then another change in the node above that. Eventually, this will change the tree's root and the block's hash, causing the protocol to register it as a completely different block (almost certainly with an invalid proof of work). There is no doubt that the Merkle tree protocol is necessary for long-term viability.

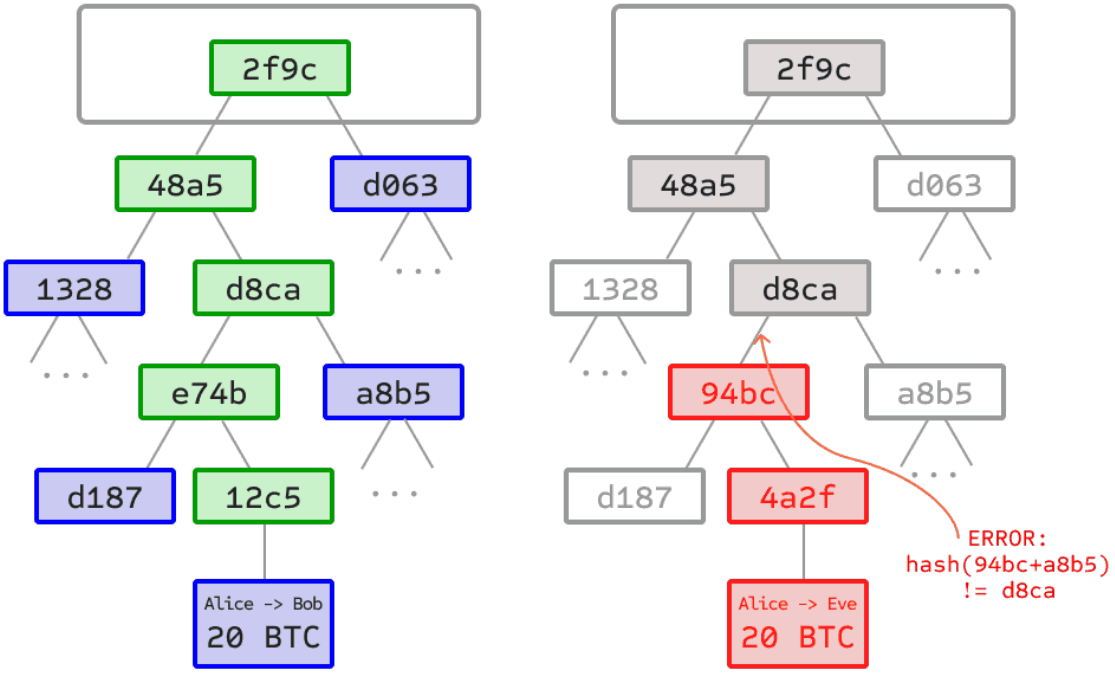

FIGURE 3.4: Merkle Tree from Ethereum white paper [1]

3.6 Consensus Mechanism

The standardized method through which the blockchain's nodes, or the computers that manage the blockchain and store the records of all transactions, reliably come to this agreement, is known as a consensus mechanism. In the field of cryptography, a consensus mechanism aims to stop bad actors from willfully cheating. The practice of "double-spending" [13] is a prime example of fraud in the crypto sphere. Users of a blockchain network adhere to this

approach to determine the validity of transactions. This technique makes sure that each copy of the blockchain contains all valid transactions and that all lawful transactions are recorded on the blockchain. On the majority of blockchains, new transactions are validated by computers known as miners. These miners compete with one another in a proof-of-work system to validate the subsequent block of transactions. The network's transaction senders fund the mining fee that the successful miner receives as payment.

Each fresh block of transactions is sent to all other miners via the consensus method, which also makes sure that all miners agree on the following block of transactions. Anyone with a device that supports nodes can download a copy of the blockchain. The ledger matches perfectly in every copy.

The consensus process makes sure there is always agreement over which assets belong in

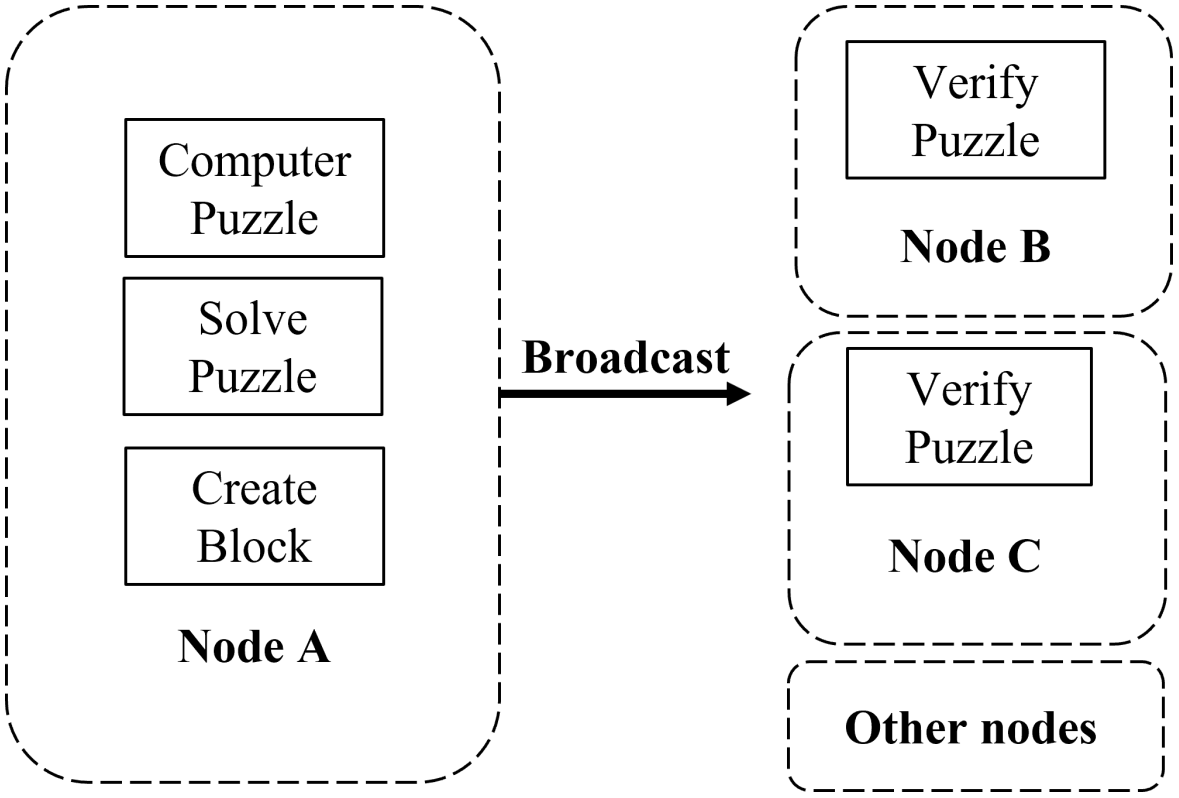

FIGURE 3.5: Proof-of-work consensus mechanism

which wallet. The two most widespread consensus mechanisms are Proof-of-work (PoW) and Proof-of-stake (PoS). Proof-of-work is a highly energy-intensive consensus mechanism but brings a high degree of trust. On the other hand, Proof-of-Stake is a consensus process

where new blocks are validated by those who hold most of the network's currency. Transactions are made possible faster and more affordably. It encourages continuing involvement by rewarding those who have the greatest stake in the network. As Proof-of-Stake is more energy-efficient than Proof-of-Work Ethereum network recently transferred from PoW to PoS reducing energy consumption in the entire world.

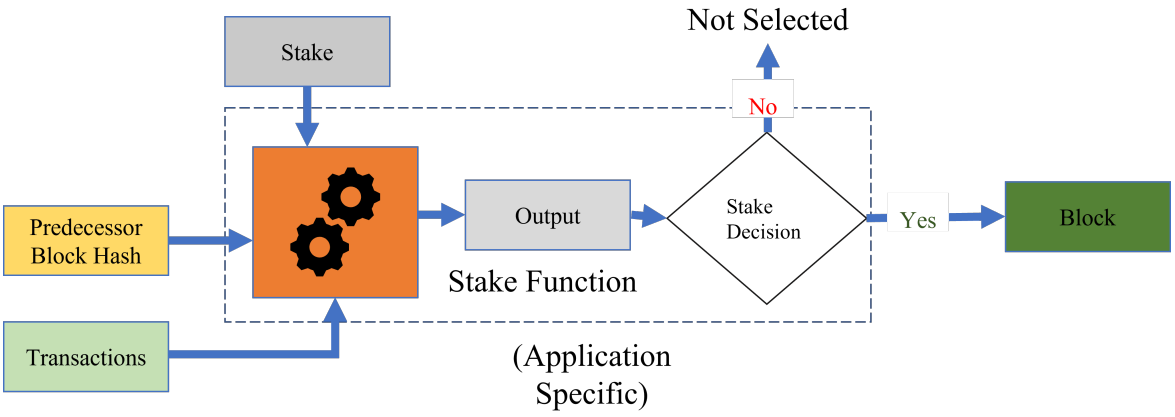

FIGURE 3.6: Proof-of-stake consensus mechanism flow [2]

3.7 MetaMask

To store tokens, engage with decentralized apps, and trade Ethereum, users can utilize the cryptocurrency wallet MetaMask, which is accessible as a browser plugin. Blockchain wallets, which are accessible as digital or online wallets, allow users to store and manage their Bitcoin, Ether, and other cryptocurrencies. Blockchain wallets facilitate bitcoin transactions, stop cryptocurrency theft, and let users change cryptocurrency holdings back into their home currencies if necessary. A browser plugin that can function as an Ethereum wallet is MetaMask. Users may use MetaMask in any browser since, unlike conventional wallets, it doesn't need any additional plug-ins. Both desktop and mobile systems can run the MetaMask program. On all supported browsers, the downloading process is substantially the same. Make sure you are aware of the tool's benefits and drawbacks before deciding to utilize it. pp-MetaMask provides a safe and practical wallet. MetaMask, an open-source program, is a dependable utility with a simple user interface and trustworthy customer service. It gives

consumers total access to and management over their local device finances. MetaMask, as its name suggests, serves as a portal into the world of dApps. MetaMask links to decentralized apps and smart contracts, in contrast to existing centralized exchanges. Events and actions may be carried out utilizing dApps. These acts often ask for payment in Ethereum or a crypto token.

Chapter 4

Proposed Methodology

4.1 Overview

In our proposed system, customers will pay their bills and tolls utilizing blockchain transactions using a web-based decentralized application (dapp). The procedure of paying a utility bill involves a third-party medium, which consumers use to pay their respective bills or tolls, however, it is not transparent and the third-party medium can be quite expensive in some situations. Therefore, we are recommending in our system that customers be able to pay their bills directly to government sources without the use of any intermediary.

The user's account, through which they will interact with the system, is the first crucial component. For each user to make a cryptocurrency payment, they must have an Ethereum wallet, in this case, MetaMask. Users will then log in to our system using their wallet, and thanks to this connection, our system may request transactions using the user's account public address via the wallet (MetaMask).

By using the stated procedure to log in, our system will have the user's account address and the amount of Ethereum they now possess, which will be utilized to request a transaction from their Ethereum wallet to the particular bill address they intend to pay the money to.

FIGURE 4.1: System Use Case Diagram

The user can type in the precise address they are sending money to manually, or they can scan the relevant QR codes for a number of distinct government service providers to obtain the public address of that provider's cryptocurrency wallet, which will be used to send money transactions to that provider's address. Therefore, no medium of any type is required for the procedure.

4.2 System Design

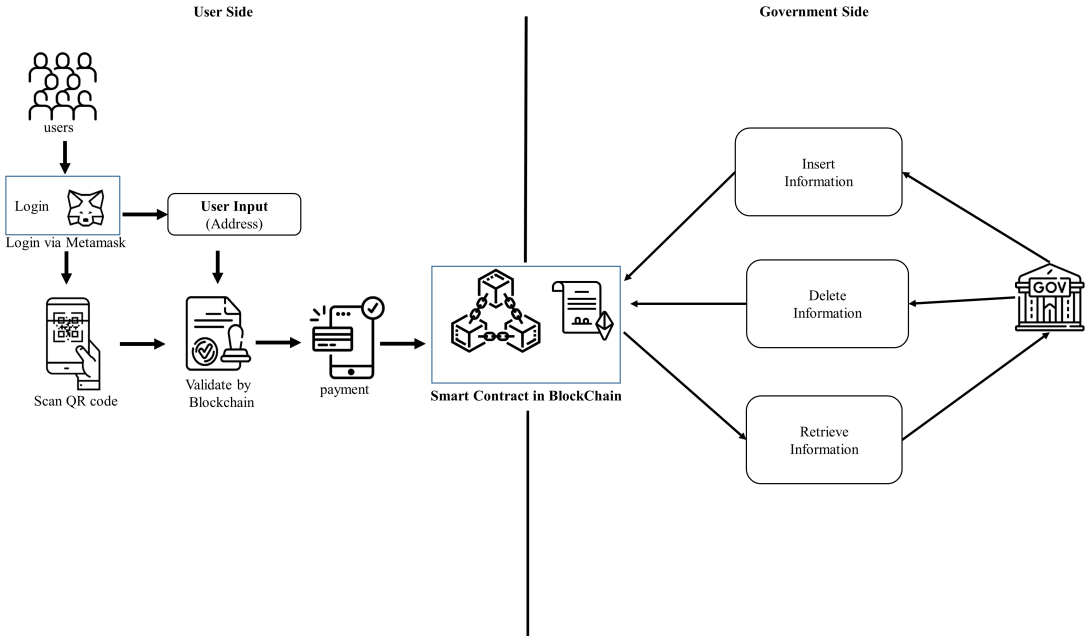

FIGURE 4.2: System Architecture

Our proposed system has two-part one user side and the other government side. Users can log in to the system and send transactions to respective government organizations for their services or toll.

4.2.1 User Side

To begin using our system, the appropriate users must first log in. Every user must have a MetaMask account in order to log in. The user can utilize our system to pay their bills after logging in. In order to pay a bill, the user must enter the address of the recipient side, which is the government entity the user wishes to send the bill to. Alternatively, the user can scan a specific QR code containing the information they require, and the address will be automatically entered. The user can then confirm the receiving address. The user may then specify the payment amount and begin the money transfer. The user's transaction is then created by the smart contract and sent to the blockchain.

4.2.2 Government Side

From their side of the system, the government can alter the pre-existing data. However, access is highly limited, and no vital data may be altered by any sort of trickery.

4.3 System Components

4.3.1 MetaMask

When a user tries to log into the system, MetaMask starts working. The system is authenticated with MetaMask. When a user wishes to log in, they just click the login button, and a window asking for their password to access their MetaMask account pops up. The extension then assists with automatic login into the system after logging into MetaMask.

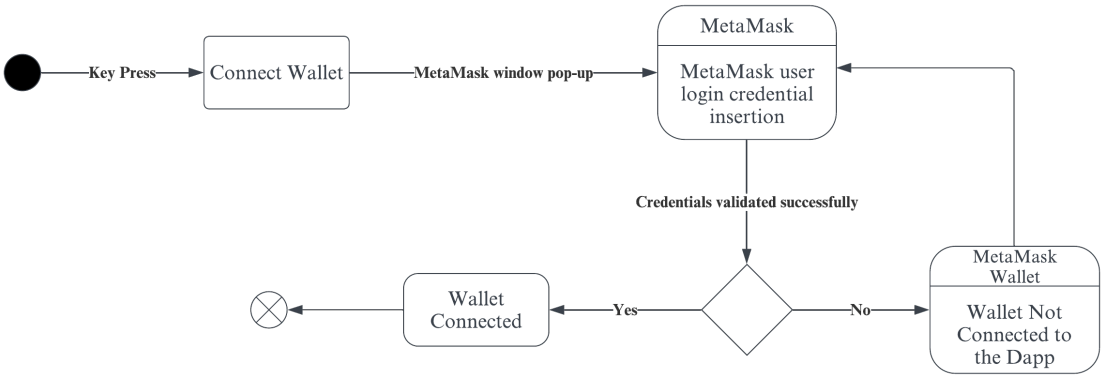

FIGURE 4.3: Login using MetaMask

The second part of work of the MetaMask starts when the transaction process is started by our systems' smart contract. A window will appear asking the user to confirm the transaction that the smart contract has requested from their wallet, which in this case is MetaMask. The user then confirms the transaction by selecting the "Confirm" button in the pop-up window, which corresponds to the user signing the transaction in the blockchain using their

private address. After MetaMask signs the transaction using the user's private address, the transaction is then sent into the blockchain to be accepted.

4.3.2 QR Code Scanner

A recipient address must be entered in order to send a transaction. We included a tool that can scan QR codes for users in order to make their work easier and more effective. Users can scan the QR codes to have the corresponding address entered for them when it comes to the account addresses of the government utility service providers. They can then confirm and begin the transaction after that.

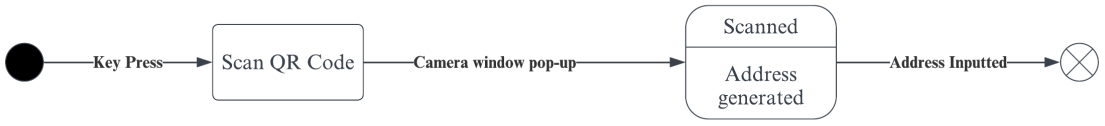

FIGURE 4.4: QR Code Scanning

4.3.3 Validation

Because an address is entered, validation is required because it is a 42-digit hexadecimal number that is difficult to remember and prone to forgetting. After pressing the verify button, the system will compare the entered address to the one provided by the government. If it finds a match, it will return positive information; otherwise, it will return negative information.

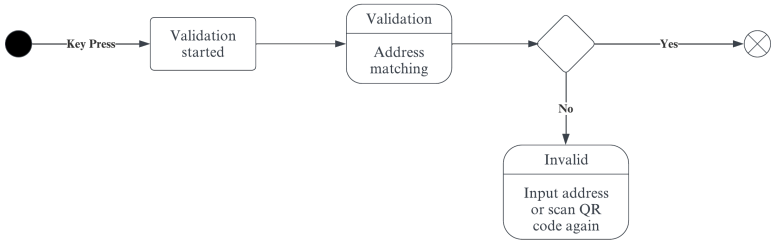

FIGURE 4.5: Validating Address

4.3.4 Smart Contract

Any blockchain application system built on Ethereum is built around smart contracts. The self-running, verifiable, nondeterministic pieces of code known as smart contracts keep the blockchain operational. Smart Contracts have the ability to store data. The majority of the labor in the blockchain is handled by our suggested model, which depends on Smart Contracts' functionality. As previously indicated, before a block is accepted, our system will send transactions to the blockchain. And a smart contract will carry out this work. Because each computation is completed by paying a small gas price, smart contracts are not overly complicated, lengthy, or filled with calculations.

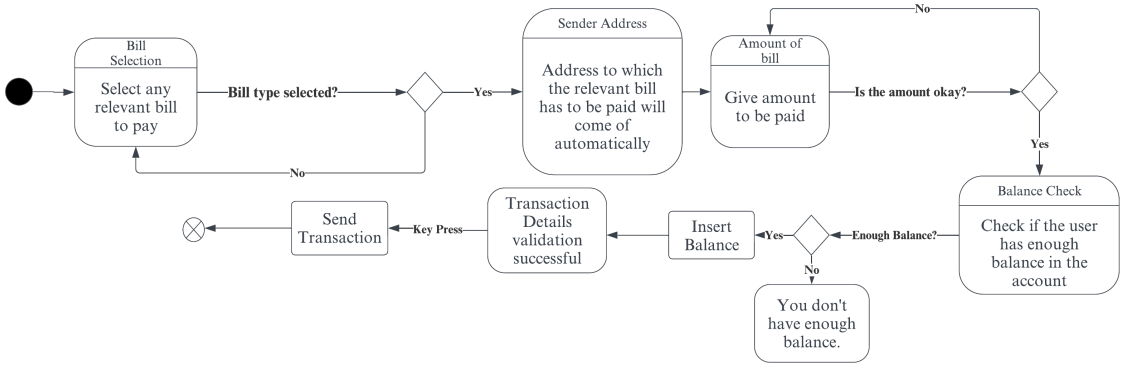

FIGURE 4.6: Transaction Process

According to the principles behind our design, a smart contract starts a transaction between two parties—one of which is a user linked to our system and the other is a relevant government service provider to whom the user would give money.

4.4 System Entities

4.4.1 Utility Service Provider

Utility service providers are mainly government organizations that provide region-wise utility services like electricity, gas, etc. People pay them for the services they provide and they

	Pseudo-code: Smart contract for validation and payment
1.	struct Establishment{address adr; string loc; string kind; int tp;}
2.	Establishment[] est; //array of structure created}
3.	procedure_insert() // insert into the structure array}
4.	procedure_isValid() // check if the address is a verified.}
5.	procedure_delete() //removes address from the array}
7.	event recipeints(of,recipient,name);}

8.	procedure_transfer(_to, symbol) public payable
	emit transactions(msg.sender, to, msg.value);
	//transfers values to another acount

9.	procedure saveTx(from, to, amount) public
	emit transactions(from, to, amount, symbol);
	//save transaction in the blockchain

TABLE 4.1: Pseudo Code of Smart Contract

also control the services. Regarding our system, these providers will have their respective addresses to which users can send money using our system, and the providers no need to hire third-party mediums, and people still can pay their bills from home easily.

4.4.2 Bill Payers

Billpayers are the users of the system who wants to pay bills for their utility services. Users or billpayers will log into the account using their crypto wallet in this case which is Meta-Mask as stated above. After they will choose their specific service provider's address and the amount of the bill, he/she needs to pay for and confirm the transaction on the MetaMask.

4.4.3 Government Organization

Government organizations are the receiver of the transaction amount. Transaction sent by the user (smart contract) after being successful the respective amount will be added to these organizations' accounts. Thus, no middleman is needed here, and the system will be cost-effective.

4.5 Implementation

Based on our proposed model after all considerations we prototyped a functional bill payment system having the entities. The prototype we devised can function as a complete bill payment system having functionalities such as transaction submission, secure and automatic transaction process, medium less system, etc. The choice of platform for developing the blockchain application is Ethereum for its open-source nature. We need the open-source nature for the latter part of our proposed method to work as a transparent publicly accessible data source. Because Ethereum is a well-known open source publicly available blockchain technology with the capability of creating and running smart contracts in a specific virtual machine, known as Ethereum Virtual Machine (EVM).

Smart Contract is written in Solidity language supported in Ethereum blockchain. Ethers are used as a cryptocurrency. The miners who run the Ethereum nodes must be paid in "Gas" for the costs of deploying and running Smart Contracts. The amount of gas consumed is determined by the transaction's computational cost. The reason for calculating the amount of gas consumed in each transaction is to pay network validators for their work securing the blockchain and network. These network validators are people or nodes on the network by whose devices' computational powers combined the network is operational. After the proof of stake algorithm was rolled out in September 2022, gas fees became the reward for staking ETH and participating in validation – the more a user has staked, the more they can earn. Verification on the network would be slow and result in a processing bottleneck if transactions were allowed to be arbitrarily complex.

Miners will utilize the cost of gas, as placed on each transaction by the node that pushed it, to assess whether or not it is worth including the transaction in the block that they are mining. Trying to push a transaction that is too complex or has a low gas cost will cause the miners to disregard it when deciding which transactions to include in their blockchain. Our smart contract does have a gas limit corresponding to the work it's done in the blockchain.

Chapter 5

Environmental Setup

We created the smart contract using Remix then we deployed the smart contract in the Gorelli Test Net. We have created our Graphical User Interface(GUI) in React.js. We have used the modules named ethers.js to interact with the smart contract. Below We have given some pictures of our graphical User Interface that we created.
First, we will be greeted with a connection to Metamask wallet. It will ask the user to connect the Crypto Wallet with the application.

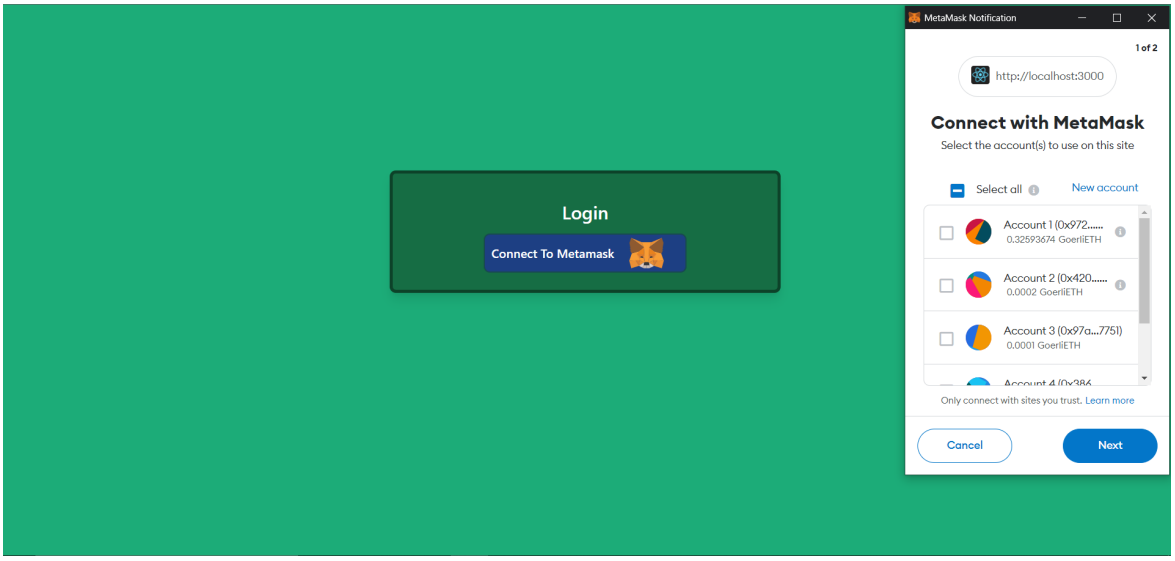

FIGURE 5.1: metamask connection

After giving the Metamask a password, Metamask will ask for permission to connect. The user must simply allow the connection.

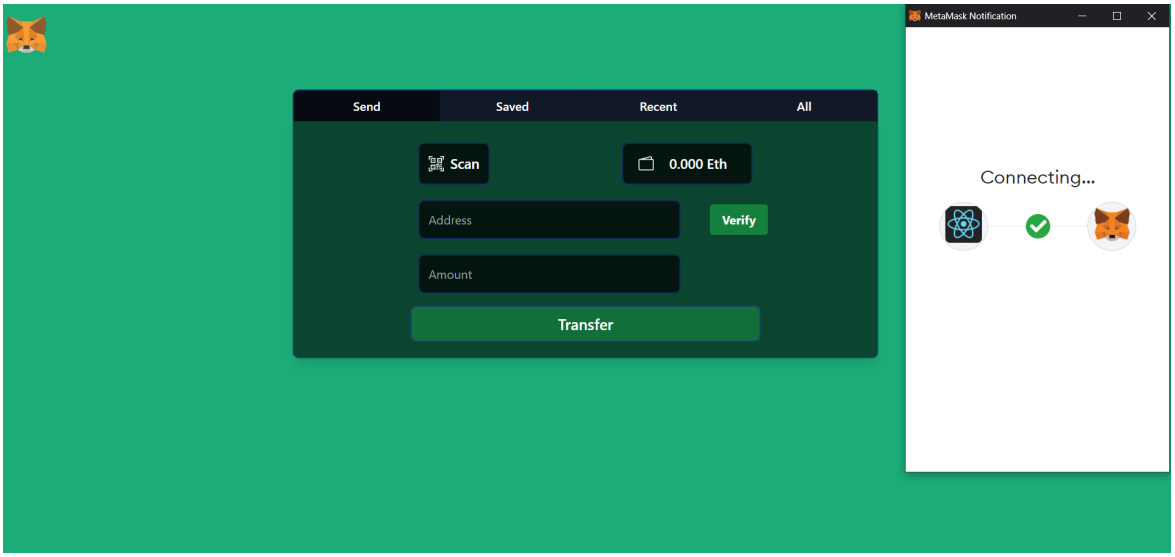

FIGURE 5.2: Metamask connected with Frontend

Then there is a button Named Scan. It allows us to scan the QR code and extract the 42-digit Metamask address.

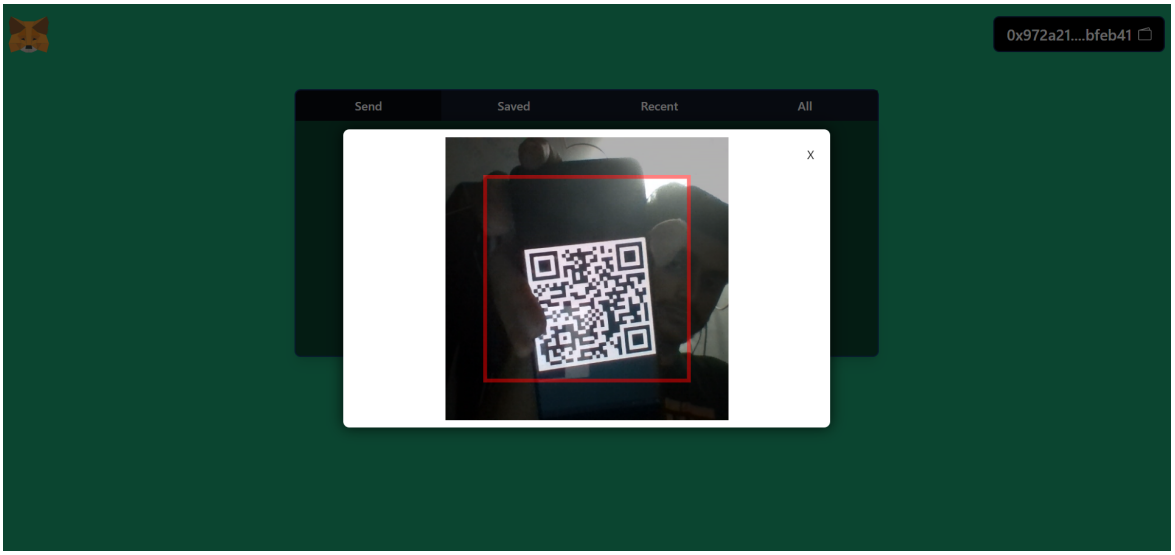

FIGURE 5.3: qr code scan

After extracting the address we have to very the address if it is legit or not. For this purpose we just simply have to click the verify button and necessary information will show up.

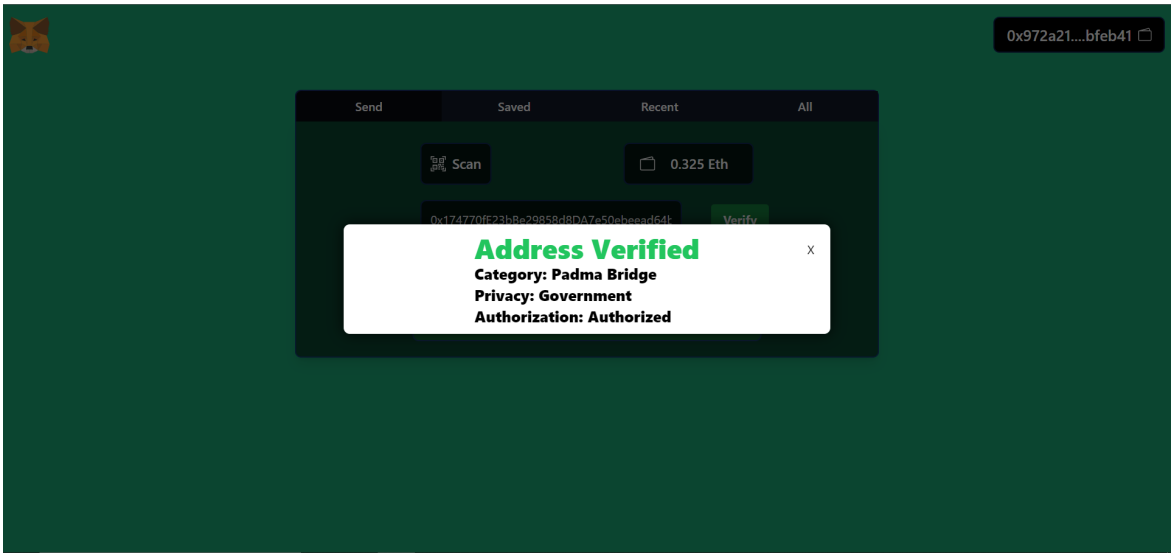

FIGURE 5.4: verification

Then we can finally transfer the balance to the Appropriate Account. After clicking the transfer button Metamask will ask for permission to Transfer the funds. We simply have to confirm the transaction.

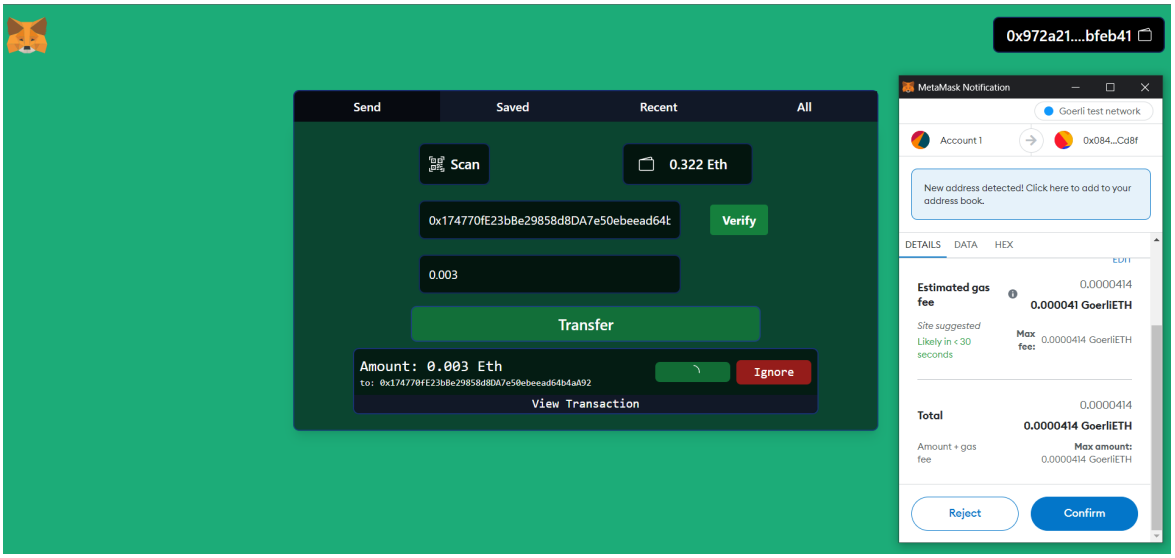

FIGURE 5.5: Transfer Balance

The recent Transactions show the sender and receiver address as well as the transfer time. One can track All the information because this Transaction information is stored on blockchain.

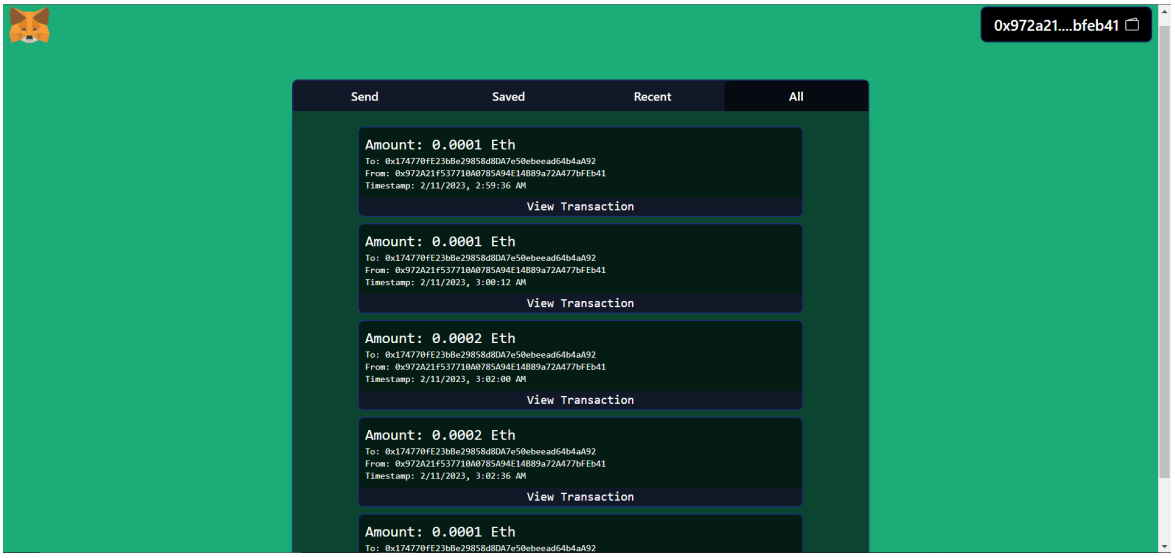

FIGURE 5.6: Recent Transactions

Chapter 6

Comparative Analaysis

Our suggested model for the government's toll and utility bill payment procedure is based on concepts from related theories in pertinent publications in the same field. The key contribution was to create something that could be used in our own national setting without going against the provisions of the constitution. enhancing any presently operating systems while at the same time implementing new ones.

- A blockchain-based application model's performance indicators are often derived from the computational cost, also known as the GAS cost of hosting the blockchain. However, since this is a cost study rather than a performance evaluation, it does not provide a whole picture of the impact. Added security is always accompanied by a little premium in price.
- To concentrate on the actual real-life effects that our suggested model may be anticipated to have, a comparison can be made with the current tender processing system to see how our model enhances specific areas.
- Corruption, extortion, and additional fees with laborious tasks might arise throughout the collection process using the standard utility bill and toll payment system. Traditionally, crooked authorities, politicians, and local thugs are responsible for this. This is not a concern with our blockchain-based architecture since once the money collection is started on the blockchain, it cannot be changed. Additionally, the transaction will take place utilizing a crypto wallet, eliminating the need for a middleman to stand between the tax payer and the collector. A whole new blockchain must be started with fresh transaction information in order to modify the details. Because there is no transparency in the process, the current approach runs the danger of affecting the money collecting process.
- Our model emphasizes this issue particularly well by providing the means of publishing the collection related information in the Smart contract.

- Any kind of cyber-attacks can alter the data of an entire collection process to hamper the entire process. A blockchain-based model is free from the risk of data alteration as the proof-of-work consensus scheme will ensure the data is never altered during the process. Because we know that in a distributed system there as to be at least 33% hacked data to temper the whole distributed system. This is known as Byzantine fault tolerance. Where is case of blockchain we have to take control of the 51% nodes to replace the whole blockchain network with a new one.

- Some officials, goons missuses their power and collect money illegally from people. But in our method the money will only go to the accounts which will be added to the blockchain by the government. This method immediately cuts downs one of the major corruption roots.

Chapter 7

Conclusions

7.1 Summary

In order to determine a recommended technique for our blockchain-based utility bill and government toll payment model, we investigated ideas related to Ethereum, smart contracts, and the usage of encryption technologies to help with access control. The suggested method made use of Ethereum smart contracts as a way to gather government toll and utility bill payments. The smart contracts for the Ethereum network were created using the computer language Solidity.

The prototyping of the model was done using tools such as Metamask, Remix, Solidity, React Js, and other frameworks. The full source code along with the smart contract can be found in our GitHub profile.

7.2 Conclusions

Change from any existing traditional system is always challenging and involves risk. And changing human lives directly by integrating newer technologies into real-life human works is the best way to use technology. A monetary collection such as payments and government toll from the recent and far past involved many blunders, political power practices,s and ridiculous amounts of frauds Transition to a newer but more secure system, although cumbersome, should be welcome nonetheless. Our work was a prototype of a solution for part of the problems. But in the bigger picture, much more work remains to be done for technology like this to be truly integrated into our day-to-day life to make life easier. We will hope we have been able to act as the foundation to inspire future works to be built on this to make it truly a reality. Blockchain is said to revolutionize the current web which is WEB2.0 where there will be no central authority that holds the utmost power. This revolution is already

happening in front of our eyes. So we must utilize these moments and create a better future for upcoming generations.

7.3 Future Works

Our attempt at prototyping the concept has been successful, and a larger-scale version of it may be incorporated into the primary toll-collecting system of our national procurements. An actual working version of this that integrates with the current digital procurement systems may be built. A more thorough investigation of the computing expenses and the extra difficulties given by a large-scale scenario would be required for this work.

We can use RFID with blockchain to make this payment process much easier, where we can use NFC to connect two people and then make payments between them using a smart contract. We also can use Machine learning to detect the number plate of the vehicle and from the stored information on the blockchain about that number plate we can automatically generate the necessary amount of toll for each vehicle.

References

[1] V. Buterin *et al.*, "A next-generation smart contract and decentralized application platform," *white paper*, vol. 3, no. 37, pp. 2–1, 2014.

[2] L. Luu, Y. Velner, J. Teutsch, and P. Saxena, "Smart pool: Practical decentralized pooled mining." in *USENIX Security Symposium*, 2017, pp. 1409–1426.

[3] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," *Decentralized business review*, p. 21260, 2008.

[4] D. Yayman, "Blockchain in taxation," *Journal of Accounting and Finance*, vol. 21, no. 4, pp. 140–155, 2021.

[5] H. Hyvärinen, M. Risius, and G. Friis, "A blockchain-based approach towards overcoming financial fraud in public sector services," *Business & Information Systems Engineering*, vol. 59, pp. 441–456, 2017.

[6] K. Wu, "An empirical study of blockchain-based decentralized applications," *arXiv preprint arXiv:1902.04969*, 2019.

[7] M. Krogsbøll, L. H. Borre, T. Slaats, and S. Debois, "Smart contracts for government processes: case study and prototype implementation (short paper)," in *Financial Cryptography and Data Security: 24th International Conference, FC 2020, Kota Kinabalu, Malaysia, February 10–14, 2020 Revised Selected Papers 24*. Springer, 2020, pp. 676–684.

[8] M. S. Setyowati, N. D. Utami, A. H. Saragih, and A. Hendrawan, "Blockchain technology application for value-added tax systems," *Journal of Open Innovation: Technology, Market, and Complexity*, vol. 6, no. 4, p. 156, 2020.

[9] H. Demirhan, "Effective taxation system by blockchain technology," *Blockchain Economics and Financial Market Innovation: Financial Innovations in the Digital Age*, pp. 347–360, 2019.

[10] A. Alkhodre, S. Jan, S. Khusro, T. Ali, Y. Alsaawy, and M. Yasar, "A blockchain-based value added tax (vat) system: Saudi arabia as a use-case," *Int. J. Adv. Comput. Sci. Appl*, vol. 10, no. 9, pp. 708–716, 2019.

[11] D. Vujičić, D. Jagodić, and S. Ranić, "Blockchain technology, bitcoin, and ethereum: A brief overview," in *2018 17th international symposium infoteh-jahorina (infoteh)*. IEEE, 2018, pp. 1–6.

[12] F. Tschorsch and B. Scheuermann, "Bitcoin and beyond: A technical survey on decentralized digital currencies," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 3, pp. 2084–2123, 2016.

[13] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An overview of blockchain technology: Architecture, consensus, and future trends," in *2017 IEEE international congress on big data (BigData congress)*. Ieee, 2017, pp. 557–564.

[14] A. A. Monrat, O. Schelén, and K. Andersson, "A survey of blockchain from the perspectives of applications, challenges, and opportunities," *IEEE Access*, vol. 7, pp. 117 134–117 151, 2019.

[15] H. Yu, Z. Yang, and R. O. Sinnott, "Decentralized big data auditing for smart city environments leveraging blockchain technology," *IEEE Access*, vol. 7, pp. 6288–6296, 2018.

[16] V. Buterin, "Ethereum: platform review," *Opportunities and challenges for private and consortium blockchains*, vol. 45, 2016.

[17] G. O. Karame, E. Androulaki, and S. Capkun, "Two bitcoins at the price of one? double-spending attacks on fast payments in bitcoin," *Cryptology EPrint Archive*, 2012.

[18] J. A. Kroll, I. C. Davey, and E. W. Felten, "The economics of bitcoin mining, or bitcoin in the presence of adversaries," in *Proceedings of WEIS*, vol. 2013, no. 11. Citeseer, 2013.

[19] Z. Zheng, S. Xie, H.-N. Dai, W. Chen, X. Chen, J. Weng, and M. Imran, "An overview on smart contracts: Challenges, advances and platforms," *Future Generation Computer Systems*, vol. 105, pp. 475–491, 2020.

[20] J. Zakrzewski, "Towards verification of ethereum smart contracts: a formalization of core of solidity," in *Verified Software. Theories, Tools, and Experiments: 10th International Conference, VSTTE 2018, Oxford, UK, July 18–19, 2018, Revised Selected Papers 10*. Springer, 2018, pp. 229–247.