2022 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI) | 978-1-6654-9529-5/22/$31.00 ©2022 IEEE | DOI: 10.1109/ACCAI53970.2022.9752568

Instigating Decentralized Apps with Smart Contracts

Nikhil, Santu Panday, Arushi Saini and Dr. Neha Gupta

Faculty of Computer Applications, Manav Rachna International Institute of Research & Studies, Faridabad
E-mail : Nikhil_2019@manavrachna.net, Pandeysantu2000@gmail.com Arushisaini61@gmail.com, neha.fca@mriu.edu.in

Abstract **Blockchain is regarded as a game-changing core technology. Blockchain is a popular technology these days, and all of the large internet businesses, such as Google, Facebook, and Amazon, are seeking for blockchain developers. Despite the fact that many researchers have achieved the benefits of blockchain, blockchain research is still in its infancy. This paper examines the current state of blockchain in India, particularly in the domain of Decentralize Applications (Dapps). The paper will discuss the importance of blockchain in digital currency and its use in development of decentralized apps across the world. The paper also discusses the concept of smart contracts which allows us to communicate with blockchain. The paper has also identified the reason of scarcity of blockchain developers in India and how it can digitally transform the future. Finally the paper will present a single-page web application using JavaScript to demonstrate the concept of decentralized apps using blockchain technology.**

Keywords—BlockChain;Smart Contract; JavaScript; Python; Ethereum, Bitcoin.

I. INTRODUCTION

Bitcoin is supported by a General Ledger Collaboratively established by its community network. Blockchain technology was originally appeared as the cornerstone of bitcoin digital currency. Developers are always working on ways to incorporate Blockchain Technology into our daily lives, and the possible uses for digital ledger technology is almost endless. Blockchains create eternal, changeless digital records and has opened new avenues for businesses to deliver Proof-of-Performance thereby establishing confidence in previously unseen digital world [10].When it comes to maintaining the track of financial assets, blockchain technology has shown to be one of the most reliable technologies. As a result of this many firms are interested in incorporating the unique features of this technology into their own security frameworks.The blockchain is only one component of a standard technological stack. To understand blockchain, engineers with networking or security expertise work alongside those with core software development skills. With blockchain technology, job opportunities are plentiful. There are more than 7000 capable blockchain developers in the planet and there is expected to be a demand for 500,000 developers by 2022.

II. SMART CONTRACT

There are several blockchains around the world, each have their own unique set of protocols for running applications. We have worked on the Binance smart chain, which is supported by Smart Contracts. A smart contract is simply a programme that is stored on a Blockchain and executes when certain conditions are met [11]. Smart Contracts are commonly used to self-start the execution of an argument so that all participants may be certain of the outcome right away, without the need for a mediator. Smart contract can also be used to automate the workflow by triggering the next step when certain circumstances are fulfilled.Smart Contract works withsome simple “if/else than” statements which are formulated inside code on a Blockchain. Computerperforms the actions whenproposed conditionare met and authenticated. This action could be anything likesending funds to the appropriate wallet, enrolling a vehicle, sending and receiving notification, or booking a ticket. After performing all the tasks, data on blockchain gets updated when the all the transactions are completed. To conduct trustworthy transactions among many parties, smart contracts do away with the need for a central authority or a mediator. All transactions are visible, safe, and traceable because smart contracts inherit all blockchain characteristics [9].A developer can programme the Smart Contract, and blockchain-based systematizations provide templates, web interfaces, and other online tools for structuring smart contracts [8].Smart contract needs to deploy on blockchain network and for doing so we used the online IDE called Remix.*https://remix.ethereum.org/*IDE is an open-source web and desktop application. It promotes a

quick development cycle and has a robust set of plugins with user-friendly interfaces. The remix is used throughout the contract development process as well as a playground for learning and teaching Ethereum [7]. This is the platform from which we can deploy our Smart Contracts and Remix convert the contract code into ABI, allowing us to use smart contract methods to read and write data on the block chain [5].

III. Decentralized Apps

Decentralized apps are intended to run on the Internet without even being supervised by a central authority and are executed on a peer-to-peer network of computers instead of just a single computer [6].Blockchain enabled users to trust decentralized applications while also addressing some of the programmes' shortcomings, such as missing nodes and virus-affected software. The deployment of a smart contract is required for decentralized apps on the block chain to function successfully.

WEB3:

Day after day, the data on internet is growing exponentially. Images, videos and all type of data is coming up in every second,thus this became a challenge that how to extract relevant data from this data Warehouse [5]. In this complex challenges Web3 tool become valuable for users in business for organizing information at large scale. Web3 is the third generation of the World Wide Web. This third generation is a decentralized internet with decentralized applications or (DApps).In the web3 era, information is decentralized and no one person or organization has control over your data.DApps are the next evolution of the web, extending the decentralized network concept to not only allow information to flow between users, but between users themselves [3].

DApps, being decentralized, are inherently self-regulating, and the information flow is managed by cryptographically secured smart contracts [4].The web3 supports a wide variety of decentralized technologies, including Ethereum, Monero, Qtum, IPFS, and many more

META MASK:

Meta Mask calls itself "your connection to the new web" and is designed to serve as a wallet and ID for blockchain-based online applications built on Ethereum. In addition to giving users a way to access crypto and NFTs, it promises a more secure and private browser experience. The only catch is that when users hold all of their data, they can't afford to lose it or give it away by themselves [2].

IMPLEMENTING SMART CONTRACTS FOR DAPPS:

The DApp smart contract is composed of a collection of writable and readable procedures (getter/setters) that are called by their unique function hash [1]. A smart contract procedure can be invoked by executing a transaction and inputting the procedure Hash code into the contract after it has been deployed.

Functions that can be written:

a) The DApp smart contract is composed of a collection of writable and readable procedures (getter/setters) that are called by their unique function hash [1]. A smart contract procedure can be invoked by executing a transaction and inputting the procedure Hash code into the contract after it has been deployed.
b) The sign function verifies that the item has been received at each checkpoint along the road until it arrives at its final destination.
c) The maintenance () method is completely optional that can be used to add further content to an item, such as "second-hand product" or "changes," as well as track its chronology.
d) The procedure changeReceiver() is used to update the beneficiary address at each security checkpoint until the product arrives at its final destination. This feature will only be enabled if the receiver indicates that he or she has received the product.

Functions that can be read:

The proposed implementation includes a number of reading functionalities, including, but not limited to:

a) seeing the information of a sent item;
b) tracking an item's position;
c) Tracking an item's maintenance history. Based on the requirements and design logic, further the fundamental implementation could be enhanced with new capabilities.

A USE CASE EXAMPLE OF DAPP:

The below mentioned use-case demonstrates the working of DApp.The below designed Lottery app is very similar to our real-world game where people play and earn money in short period of time, and is implemented on blockchains. PancakeSwap lottery

is the best exampleof Dapps on blockchainbut in pancakeswap lottery appthere are multiple winners and the final winner is selected on the basis of percentage. The proposed lottery Dapp is different and the winner will be select using random number.Before reading and writing data on block chain frontend is designed using JavaScript and Veu.js and WEB3.js has been used for reading the Smart Contract ABI.

IMPLEMENTATION OF LOTTERY DAPP:

a) Initializing Web3

Before writing any blockchaincode web3 is initialized&Meta mask is installedon web browser as shown in figure 1.

```
import Web3 from "web3";
var web3 = null;
if (window.ethereum) {
  web3 = new Web3(window.ethereum);
}

export default web3;
```

Figure 1: Initialization of web3

First of all web3.js is imported from node modules so that we can use web3 object. A null variable is initialized to check if our browser support communication with blockchain or not and if it does than weinitialize anEthereum object from web3.

b) Creating Frontend

To make our application user friendly, we have designed a frontend for the lottery app and from this frontend we'll be able to interact with our smart contract which will act as a bridge to connect with blockchain on BCS network. We have used Vue.js Framework for creating the frontend.

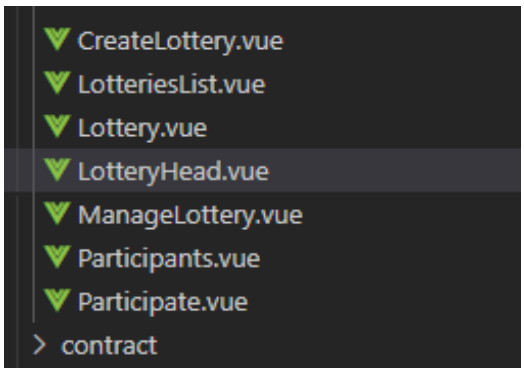

Figure 2: Creating frontend using veu.js

We have coded our frontend in these veu.js file as shown in figure 2 and we will be using these file as the components of our application

c) Connecting with blockchain

Next task is to connect the frontend with block chain as shown in figure 3 and for this we have usedweb3 and setup the network on Binancechain in Meta mask.

```
import web3 from './web3';
import lotteryJson from '../contract/Lottery.json'

var lottery;
if (web3) {
    lottery = new web3.eth.Contract(JSON.parse(lotteryJson.interface));
}

export default lottery;
```

Figure 3: Connecting with blockchain

We are reading the contract methods usingBI with the help of web3 object and save the new contract object into the new lottery variable from which we can easily call the methods of contract as shown in figure 4.

```
Lottery.methods
  .ethToParticipate()
  .call()
  .then(result => {
    this.ethToParticipate = result;
  });
Lottery.methods
  .maxEntriesForPlayer()
  .call()
  .then(result => {
    this.maxEntriesForPlayer = result;
  });
Lottery.methods
  .getWinningPrice()
  .call()
  .then(result => {
    this.currentWinningPrice = web3.utils.fromWei(
  });
Lottery.methods
  .getPlayer(this.accounts[0])
  .call()
  .then(player => {
    this.player = player;
```

Figure 4: Reading the Contract

Methods of contract with lottery object and web3objects have been called using different component file so that each method will execute whenthe particular component will render on screen.Finally the application will run with yarn start

command andweb3 will start working as shown in figure 5and sendwill send a call to Meta mask to access the account.

```
import Header from "./components/shared/Header";
import Footer from "./components/shared/Footer";

import web3 from "./web3/web3";

export default {
  data() {
    return {
      isMetaMaskPresent: false,
      isMetaMaskLoggedin: false,
    };
  },
  async created() {
    this.isMetaMaskPresent = web3 ? true : false;
    try {
      await ethereum.enable();
    } catch (error) {
    }
    if (this.isMetaMaskPresent) {
      const accounts = await web3.eth.getAccounts();
      this.isMetaMaskLoggedin = accounts.length ? true : false;
    }
  },
  components: {
    AppHeader: Header,
    AppFooter: Footer
  }
};
```

Figure 5: Running the application

Now our Dappis ready with all the methods and components. If a browser does not support Ethereum connection the app will show a warning saying “Please install metamask first”. We have also used exception handling for non-Ethereum web browser, and this will not let our application stop while running on blockchain.

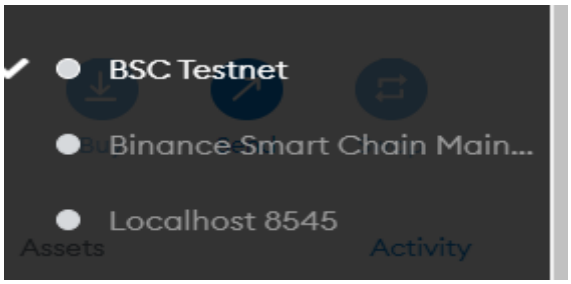

Figure 6: BSC test net

As the Smart Contract is deployedin Binance Smart chain test net as shown in figure 6, we also need to switch in BSC test net. After that Owner will be able to create lottery as shown in figure 7.

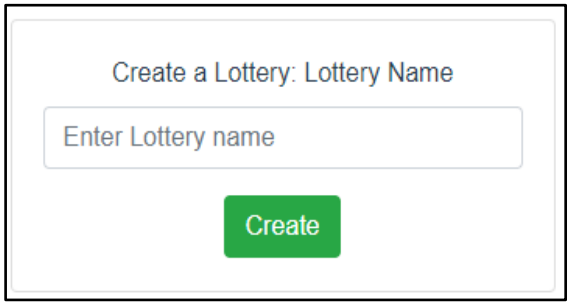

Figure 7: GUI of Lottery app

Now user can participate in the lottery as shown in figure 8 and can win prizes using the app.User will be paying 0.01 BNB for entering into the lottery and after time out Owner of the Smart Contract will declare the winner and all the winning amount will be sent to winner wallet address by smart contract.

Figure 8: Participation in Lottery app

To evaluate the efficiency of app we have calculated the execution gas value & execution time of each function used in deployment of smart contract of the DApp. Table 1 outlines the gas values used for calling each function.

Table 1: Estimated execution gas per function

Function	**gas limit**	**gas price (Gwei)**
sendProduct()	243592	4
sign()	793692	31
maintenance()	160712	31
changeReceiver()	35735	31

To highlight the important executional changes, we employed different gas values for each function of the proposed smart contract as shown in table 2. We can calculate the maximum transaction fee necessary for each function to be executed by multiplying the gas limit by the gas price on the Ethereum network.

Table 2: Execution time per function

Function	execution time (sec)
contract deployment	< 17
changeReceiver()	< 14
maintenance()	< 5
sign()	< 9

Execution time is no longer an issue in the Ethereum network because the Gas values may be modified. We have identified that that the Maintenance function has the larger cost; however, we can see from Table 2 that this function only takes 5 seconds to verify on the Blockchain. The remaining functions utilized values suggested by the Ethereum network or altered by us, and they too took a few seconds to verify.

IV. Conclusion

After developing this Dapp, we can confidently state that Block Chain is the most promising technology for security and storage in the future. People began to decentralise their applications. Because of the growing demand for developers, Block Chain has also become the best source of income. In recent years, this technology has advanced at a rapid rate in terms of advancement and widespread adoption, with no signs of slowing down. Bitcoin, Ethereum, and BNB are examples of blockchain currencies that are constantly growing as new blocks are added to the chain, significantly increasing the security of the ledger. This means that now is the time to prepare for a financial and social revolution that will change the way data is managed.Each transaction is recorded in an unchangeable and changeless manner using blockchain technology. Deception, hacking, data fraud, and information loss are all tough with this incomprehensible digital ledger.Blockchain technology will have an impact on every industry on the planet, including manufacturing, retail, transportation, healthcare, and real estate. It will make life easier and safer by modernizing the collection of personal data.

References

[1] Yadav, Arun, Divakar Yadav, Sonam Gupta, Dharmendra Kumar, and Pankaj Kumar. "Online food court payment system using blockchaintechnolgy." In 2018 5th IEEE Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON), pp. 1-7. IEEE, 2018.

[2] Chakraborty, Sabyasachi, SatyabrataAich, and Hee-Cheol Kim. "A secure healthcare system design framework using blockchain technology." In 2019 21st International Conference on Advanced Communication Technology (ICACT), pp. 260-264. IEEE, 2019.

[3] Baptista, Gonçalo, and Tiago Oliveira. "Understanding mobile banking: The unified theory of acceptance and use of technology combined with cultural moderators." Computers in Human Behavior 50 (2015): 418-430.

[4] Gupta, Neha. "A Deep Dive Into Security and Privacy Issues of Blockchain Technologies." In Handbook of Research on Blockchain Technology, pp. 95-112. Academic Press, 2020.

[5] Gupta, Neha. "Security and privacy issues of blockchain technology." In Advanced Applications of Blockchain Technology, pp. 207-226. Springer, Singapore, 2020.

[6] Shandilya, Akash, Himanshu Gupta, and Sunil Kumar Khatri. "Role and Aplications of Iot in Online Transactions using Blockchain Technology." In 2018 International Conference on Advances in Computing and Communication Engineering (ICACCE), pp. 465-470. IEEE, 2018.

[7] Sakho, Seybou, Zhang Jianbiao, FirdaousEssaf, and Khalid Badiss. "Improving Banking Transactions Using Blockchain Technology." In 2019 IEEE 5th International Conference on Computer and Communications (ICCC), pp. 1258-1263. IEEE, 2019.

[8] Chakraborty, Sabyasachi, SatyabrataAich, and Hee-Cheol Kim. "A secure healthcare system design framework using blockchain technology." In 2019 21st International Conference on Advanced Communication Technology (ICACT), pp. 260-264. IEEE, 2019.

[9] Hon, W. K., John Palfreyman, and Matthew Tegart. "Distributed ledger technology & Cybersecurity." European Union Agency For Network And Information Securit (ENISA) (2016).

[10] Ahn, Jaehong, Mingyu Park, and JeongyeupPaek. "Reptor: A model for deriving trust and reputation on blockchain-based electronic payment system." In 2018 International Conference on Information and Communication Technology Convergence (ICTC), pp. 1431-1436. IEEE, 2018.

[11] Guidi, Barbara. "When blockchain meets online social networks." Pervasive and Mobile Computing 62 (2020): 101131.