



Blockchain Creation Using Java Programming Language

Dr. Santosh Kumar Singh¹, Dr. Varun Tiwari², Dr. Vikas Rao Vadi³

Assistant Professor, Department of CS & IT, DBIT, GGSIPU, New Delhi, India¹

Associate Professor, Department of CS & IT, DBIT, GGSIPU, New Delhi, India²

Professor, Director, DBIT, GGSIPU, New Delhi, India³

*Corresponding Author: santosh.trinity17@gmail.com

Abstract: Blockchain developing in the commercial and digital world, and individuals are interested to pick up this technology. It is entering all domains very rapidly. We can't disagree that blockchain is profitable due to its firm, safe, and distributed platforms. It has an optimistic perspective. Professionals in different domains are applying blockchain. Blockchain creation is nowadays in requirement of dedicated developers of blockchain as new and more assignments are being ongoing in the blockchain platform. Lots of developers shifting their concentration in the direction of blockchain technology. For S/W developers to move in the blockchain region, they require to have a deep understanding of the languages used for blockchain programming. For mounting blockchain-driven applications and systems, developers use programming languages like Java, Solidity, Rust, JavaScript, Python, C++ and so many. In this research work, we will get to know for blockchain development about the utmost used programming languages i.e. Java. As well as we will get to be acquainted with the advantages, disadvantages, features, and explanations for seeing the Java language in blockchain technology. Further in this research work, we will display how to get the hash value of any string after that block creation as well as blockchain creation. During this Blockchain creation, we will display the hash value of the genesis block and the hash value of the 2nd block. The Java application is proposed to be used in teaching space surroundings by mentors when they demonstrate preliminary Blockchain progresses, which is the main objective of this research paper.

Keywords: Block, Blockchain, Hash, Java, Security.

I. INTRODUCTION

[The digital revolution includes the usage of the most recent information technologies to facilitate important developments in every business domain. These innovative technologies are generating new openings, on the other hand also permitting real revolution of commercial technologies already in place [1]. Blockchain technology is created for distributed and irreversible information storage. In the beginning, this technology is functional to hold the track of monetary dealings in a system of a blockchain i.e. distributed ledger. The digital ledger arrangements convert transactional information into a file that stores unalterable data i.e. blocks within a definite interval, and at that time links these blocks to generate a chain structure [2]. Node, Hash, Mining, and Bitcoin are key terms associated with the digital ledger. A node can be described as a minor server where the data is stored. Hash represents using a mathematical function to create a value from an input value. Mining is a process where new bitcoins are created and during creation authenticate all phases in the dealings when in service of bitcoins or any other crypto-currencies. Bitcoin ideas and types of machinery produce the foundation of a virtual or digital currency environment [3]. Blockchain technology, composed of the (IoT) and (AI), is understood as one of the key drivers for the revolution and digital revolution [4]. Blockchain machinery is gaining impetus for several application areas in various domains.

Satoshi Nakamoto introduced Bitcoin in 2009 and the introduction of Ethereum tokens in 2015, since then, the improvement of the digital ledger has not ever been immobile. Some developments will dominate along with blockchain technology like BaaS, IoT, AI, Social networking, and distributed investment. The inspiration for this exploration originates from the information that distributed ledger technology is acknowledged as very encouraging for several applications in the commercial area such as in *Securities* (Derivatives, Debt, Crowdfunding, Private Markets, Equity), in *Smart Contracts* (Digital Rights, Wagers, Escrow), in *Digital Currency* (E-commerce, Microfinance, P@P Lending, Global Payment, Remittance), and *Record-Keeping* (Healthcare, Intellectual Property, Voting, Ownership) [5]. In this work of research, we emphasize putting into practice a distributed ledger using the open-source platform java [6]. Such a system is deliberated for promising use in teaching space.



The rest of the paper is prepared as follows: In Section 2, we describe Java for blockchain technology as well as the implementation tool Java. Section 3 covers the steps required to generate a chain of blocks and the evaluation of hash value results. A conclusion is given at the last.

II. JAVA PROGRAMMING LANGUAGE FOR BLOCKCHAIN

When we listen to the word 'Java,' the things which come to our notice are trustworthy, robust, plus secure, and all said are unconditionally correct. Java is extremely widely held & adaptable which assistances design in Ethereum users and tools to usage in manufacturing surroundings. Java is the finest language particularly for big applications, for example, an Ethereum client for the reason that of its quickness, maintenance, and extensibility. Tools like (JMX is a tool for monitoring and handling Java applications) or (Java Mission Control, which consists of Java Management Console) support effortlessly handling applications. Java is an object-oriented programming language assistance in writing extremely moveable codes which are also used in blockchain. It is best known for "write once run anywhere" functionality i.e. we can execute the code on every gadget having a (JVM-java virtual machine). Hence, to generate a simple and unchallengeable platform for the distributed ledger this application is used [7].

Features of Java i.e. Simple, Architecture-neutral, Platform Independent, Movability, Robustness, Amazing performance, Distributed, Vibrant, Multithreaded, and Interpreted make it a practical choice for emerging distributed ledger. *Reasons for Selecting Java* for emerging Distributed blockchain apps are *Java is fast* as Ruby, Python, Go, or C through its just-in-time compiler and modern (GC) Garbage Collection algorithm. *Easy Upkeep* Ethereum-specific motives for using Java as Java is static, typed language that protects time and power which maintaining code is made easier.

The JVM supports code movability and flexibility that support all platforms for example desktop, mobile, and embedded. *Java also Backings Pluggability* as for anything it has a library, containing cryptographic concepts that might be in employment. Java appears to be the accurate choice for the Ethereum distributed ledger i.e. blockchain, and it supports drive Ethereum acceptance in the enterprise interplanetary. Java is a key language in the blockchain domain and for Java developers of blockchain equally, the demand for them is simply rising greater. *Applications in Blockchain Technology* is (IOTA) which is used to design the Internet of Things (IoT). Also, Java is implemented in the new economy movement (NEM) i.e. XEM is the native cryptocurrency of the blockchain project. Java is also used in Ethereum, IBM Blockchain, Hyperledger's contracts, and Bitcoinj [8].

III. IMPLEMENTATION OF BLOCKCHAIN

The first step in the implementation is to find the hash value. As shown in Figure 1.

```
import java.util.Arrays;
import java.util.ArrayList;
class sks
{
public static void main(String args[])
{
String[] List1 = { "a", "b"};

System.out.println(Arrays.hashCode(List1));
}
}
```

```
Command Prompt
C:\jdk-17.0.1\bin>javac sks.java
C:\jdk-17.0.1\bin>java sks
4066
C:\jdk-17.0.1\bin>
```

Figure 1. Hash Value



In a second step, we have created a block. As shown in Figure 2.

```
import java.util.Arrays;
public class Block
{
    public int previousHASH;
    private String[] transactions;
    private int blockHASH;
    public Block( int previousHASH, String[] transactions)
    {
        this.previousHASH=previousHASH;
        this.transactions=transactions;
        Object[] contens = { Arrays.hashCode(transactions), previousHASH };
        this.blockHASH = Arrays.hashCode(contens);
    }
    public int getPreviousHASH()
    {
        return previousHASH;
    }
    public String[] getTransactions()
    {
        return transactions;
    }
    public int getBlockHASH()
    {
        return blockHASH;
    }
}
```

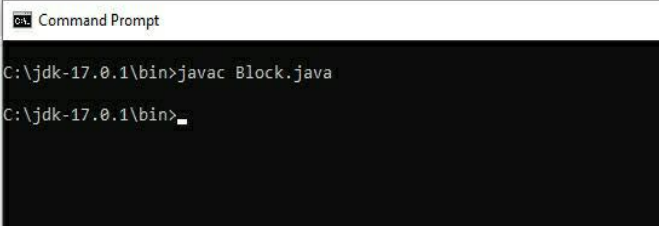


Figure 2. Block

In the third step, we have written a code in java to get the hash value of the Genesis block i.e. hash value of the first block. As shown in Figure 3.

```
import java.util.ArrayList;
public class santoshBlockchain
{
    ArrayList<Block> blockchain = new ArrayList<>();
    public static void main (String[] args)
    {
        int previousHASH = 0;
        String[] genesisTransactions = {"Santosh Singh sent to Dr. Vadi 10000 bitcoin" , "& Dr. Vadi sent 20000 bitcoin to Santosh"};
        Block genesisBlock = new Block(previousHASH, genesisTransactions);

        System.out.println("Hash of Genesis Block");
        System.out.println(genesisBlock.getBlockHASH());
    }
}
```

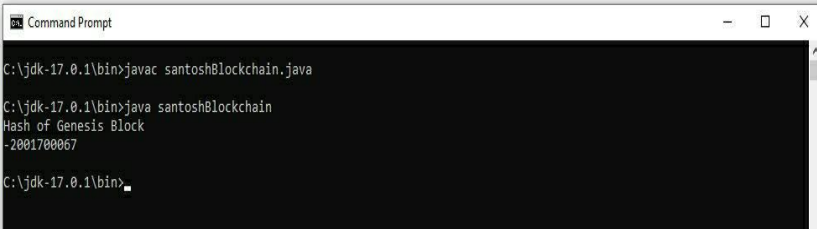


Figure 3 hash value of Genesis block

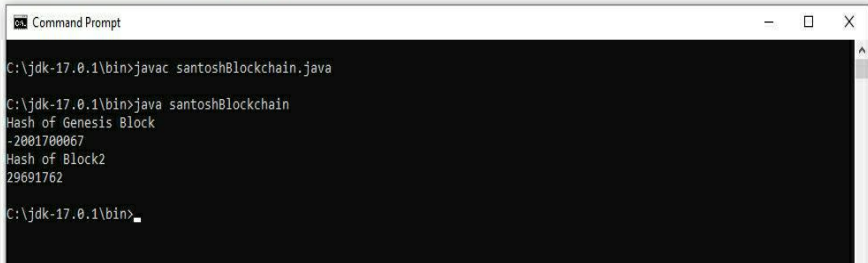


In the fourth step, we have written a code in java to get the hash value of the Genesis block as well as the hash value of the 2nd block. As shown in Figure 4.

```
import java.util.ArrayList;
public class santoshBlockchain
{
    ArrayList<Block> blockchain = new ArrayList<>();
    public static void main (String[] args)
    {
        int previousHASH = 0;
        String[] genesisTransactions = {"Santosh Singh sent to Dr. Vadi 10000 bitcoin", "& Dr. Vadi sent 20000 bitcoin to Santosh Singh"};
        Block genesisBlock = new Block(previousHASH, genesisTransactions);

        String[] block2Transactions = {"Dr. Vadi sent 20000 bitcoin to Santosh Singh", "& Santosh Singh sent to Dr. Vadi sent 10000 bitcoin "};
        Block block2 = new Block(genesisBlock.getBlockHASH(), block2Transactions);

        System.out.println("Hash of Genesis Block");
        System.out.println(genesisBlock.getBlockHASH());
        System.out.println("Hash of Block2");
        System.out.println(block2.getBlockHASH());
    }
}
```



```
C:\jdk-17.0.1\bin>javac santoshBlockchain.java
C:\jdk-17.0.1\bin>java santoshBlockchain
Hash of Genesis Block
-2001700067
Hash of Block2
29691762
C:\jdk-17.0.1\bin>
```

Figure 4 Hash value of Genesis block and the hash value of 2nd block

IV. CONCLUSION

This paper refers to a research determination aiming at the usage of blockchain technology for feasible applications in a different sector. The research paper provides the background of Java used in blockchain technology and explains an implementation using the Java open-source platform.

In this research work, first of all, we generated the hash value of any string then we created block, genesis block, and second block as well as produced the hash values of the first block i.e. genesis block then also the hash value of 2nd block and similarly we can generate several blocks which will become a blockchain.

All the required processes were executed correctly using commands in the terminal. The Java application is proposed to be used in teaching space surroundings by mentors when they demonstrate preliminary Blockchain progresses i.e. the main objective of this research paper.

REFERENCES

- [1] Maheshwari, Anup. *Digital Transformation: Building Intelligent Enterprises*. John Wiley & Sons, 2019.
- [2] Zhang, J., Zhong, S., Wang, T., Chao, H. C., & Wang, J. (2020). Blockchain-based systems and applications: a survey. *Journal of Internet Technology*, 21(1), 1-14.
- [3] Antonopoulos, A. M. (2017). *Mastering Bitcoin: Programming the open blockchain*. "O'Reilly Media, Inc."
- [4] Kim, S., Deka, G. C., & Zhang, P. (2019). *Role of blockchain technology in IoT applications*. Academic Press.
- [5] Tiscini, R., Testarmata, S., Ciaburri, M., & Ferrari, E. (2020). The blockchain as a sustainable business model innovation. *Management Decision*.
- [6] Liu, X. (2018, November). A small java application for learning blockchain. In *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)* (pp. 1271-1275). IEEE.
- [7] Svenson, C. (2017). Blockchain: Using cryptocurrency with java. *Java Magazine, January/February*, 36-46.
- [8] Singh, S. K., Manjhi, P. K., & Tiwari, R. K. (2021). Cloud Computing Security Using Blockchain Technology. In *Transforming Cybersecurity Solutions using Blockchain* (pp. 19-30). Springer, Singapore.

**BIOGRAPHY**

Dr. Santosh Kumar Singh is an Assistant Professor at Don Bosco Institute of Technology, GGSIPU, New Delhi. He has done Ph.D. from Vinoba Bhave University, Jharkhand. He has over 18 years of teaching experience and published more than 20 research papers in IEEE, Springer, and Scopus-indexed journals. His research work has been listed in H-INDEXTM as well as in I-INDEXTM also. He has written a book on Data Structures and guidebooks on Artificial Intelligence and Web Based Programming (PHP) published by AKASH Books, New Delhi. He is also serving as an academic counselor of (MCA/BCA) at IGNOU since 2006. In 2020 he was appointed as a reviewer at IGI Global Scopus Indexed Journals. He has reviewed the syllabus course titled "Web Technology" Amity University Greater Noida, and also reviewed the Springer Singapore book titled " Transforming Cyber Security Solutions Using Blockchain Technology". He was the primary evaluator of Toyathon 2021, an initiative of the Ministry of Education's Innovation Cell. He is a member of (IAENG) the International Association for Engineers, Hong Kong. His research interest areas are Blockchain, Cloud Computing, & Parallel and Distributed Computing.