International Conference on Smart Data Intelligence
(ICSMDI 2021)

Blockchain based Tamper Proof Certificates

Ms. Shraddha S. More[a], Mr. Niraj Patel[b], Mr.Sukaji Parab[c], Mr. Sushil Maurya[d]

[a]*Assistant Professor, Department of Information Technology, St. John College of Engineering and Management Palghar-401404, India.*
[b]*U.G. Student, Department of Information Technology, St. John College of Engineering and Management Palghar-401404, India.*
[c]*U.G. Student, Department of Information Technology, St. John College of Engineering and Management Palghar-401404, India.*
[d]*U.G. Student, Department of Information Technology, St. John College of Engineering and Management Palghar-401404, India.*
[a]*shraddham@sjcem.edu.in*
[b]*pniraj657@gmail.com*
[c]*parabsukaji@gmail.com*
[d]*sushilmaurya1174034@gmail.com*

A B S T R A C T

Document is the main proof for showing identity of a person and nowadays there has been many cases related to document tampering. Most of the documents like marksheet, birth certificate, identity card, etc. can be easily tampered and produced, sometimes the tampered document looks much more real and it becomes difficult for the organization or the verifier to verify the document. This research paper will try to overcome the issues with the help of blockchain technology. Blockchain is a decentralized peer-to-peer mechanism that provides protection, trustworthiness, legitimacy, immutability, and transparency. In this proposed system hashing technique is used for encryption. By using this technology people can request for the necessary documents and verifier can easily verify the documents using hash value.

Keywords — Blockchain, Digital certificate, Security, SHA-256, Certificate authentication, Digital innovation, Hash functions.

1. Introduction

Imitation of document is a developing issue and requests the most extreme consideration. The deceitful proliferation of certificates has expanded essentially as of late, turning into a hazard for partners in the instruction strategy. As per a new review by UK's National Qualification Agency (NQA), it tracked down that just one out of four college affirmation staff feel arduous spotting fake qualification documents [1]. It additionally raises worry of national security for instance the genius behind the September 13 sequential impacts in 2008 utilized the fake testament to get confirmation in one of the eminent colleges in India [2]. Therefore the validation of certificate is necessary for nation and organization security. Therefore in view of this facts we developed a block chain based certificate validation system. Blockchain is peer-to-peer decentralized system that provides security, reliability, authenticity, immutability and transparency [3]. In existing system the validation of certificate is done by checking the sign or stamp but nowadays this stamp and sign can be copied and the certificate can be tampered. So in order to overcome that problem we used hashing algorithm with association of blockchain technology so the block hash is used for the validation of certificate. The data of multiple transactions is stored in the form of blocks along with its timestamp, each transaction can be separately verified by using its hash value. SHA 256 algorithm is used in proposed system for hashing [4]. SHA 256 is one of the hash function of SHA-2 Family. The Secure Hash Algorithm 2 (SHA-2) is a series of cryptographic hash functions developed by the National Security Agency of the United States (NSA) [5]. This system can also be used in government related documents such as birth certificate, ration card and identity card etc.

The remainder of this paper is as follows. The literature that is analyzed for the work is discussed in Section 2. Section 3 discusses the SHA-256 algorithm, while Section 4 focuses on proposed and developed systems for certificate issuance and validation. The emphasis in Section 5 is on implementation. Discuss the findings and discussion in section 6. The research is concluded in Section 7, which defines the reach and possible directions.

2. Literature Survey

In 2017, Deepayan Bhowmik and Tian Feng [6] presented a system that is disseminated and sealed media transaction framework dependent on blockchain technology. Blockchain is a moderately new and promising innovation that can possibly acquaint straightforwardness and trust with transparently ensure

an arrange and approve transactions. Current media conveyance doesn't save self-retrievable data of transaction trails or substance change accounts. This paper proposes a joint physical and application layer security framework that misuses the security limit and sign preparing methods at the actual layer; and the confirmation and watermarking procedures at the application layer.

In 2019, Maharishi Shah and Priyanka Kumar [7] presented a system that is produced for effective innovation to store birth records which can't be tampered just as simple to keep up, very much dependable and effectively shareable. These procedures are likewise used to supplant passwords, pins, smartcards, keys and tokens which are the methods for verification. This system is implemented on local blockchain network using public and private keys and RSA algorithm is used for user login and registrations.

In 2019, Kumavat et al. [8] presented a system that is based on blockchain technology. The system's application is customized on the Ethereum stage and is controlled by the EVM (Ethereum Virtual Machine). In the system, three group of clients are included, Schools or colleges that give declarations, will approach the system, and will actually want to access the system database. At the point when verifier satisfy certain prerequisites, the organization will give an endorsement through the system. After the user have gotten their endorsement, they will actually want to ask about any authentication they have acquired. The specialist organization will be answerable for system maintenance.

In 2019, Malik et al. [9] presented a system that is blockchain based answer for checking the legitimacy of the certificate given by the Indian Government specialists is talked about. The upside of utilizing this system is that it gives a fast, solid and secure channel for giving specialists to get to records of a person who has different archives straight forwardly from the databases of the other giving specialists. This entrance is permissioned and time limited, so security concerns are disposed of. In this paper IFPS is utilized. IPFS (Interplanetary File System) is a distributed file putting away and sharing system that contains a few correspondences conventions in a decentralized circulated system.

In 2019, Kumar et al. [10] presented a system that is decentralized and circulated network where the user information is put away in type of blocks. These blocks are associated with each other framing a chain of records. The system is carried out utilizing Hyperledger fabric network since this is a kind of system that has a place with association so data ought to be in private mode. There are different networks like Ethereum, however they are utilized to carry out systems which are identified with public information.

3. SHA-256 Algorithm

SHA-256 is an individual from the SHA-2 cryptographic hash capacities planned by the NSA [11]. SHA represents Secure Hash Algorithm. Cryptographic hash capacities are numerical activities run on advanced information; by looking at the figured "hash" (the output from execution of the algorithm) to a known and expected hash esteem, an individual can decide the information's respectability. A single direction hash can be produced from any piece of information, however the information can't be created from the hash. Secure interchanges for sites and web administrations depend on records known as declarations. They are utilized to set up and verify secure associations. These endorsements contain cryptographic components that are produced utilizing algorithms, for example, SHA-256. The underlying adaptation of the SHA-256 algorithm was made by the US National Security Agency in the spring of 2002. A couple of months after the fact, the public metrological University distributed the recently declared encryption Protocol in the FIPS PUB 180-2 secure information handling standard received at the Federal level. In the colder time of year of 2004 it was recharged with the second form of the algorithm

Throughout the following 3 years, the NSA gave a second- generation SHA patent under royalty-free license. This is the thing that led to the utilization of innovation in regular citizen territories. This Protocol works with data separated into bits of 512 pieces (or 64 bytes as such). It creates its cryptographic "mixing" and afterward gives a 256-digit hash code. The algorithm incorporates a moderately straightforward round, which is rehashed multiple times [4].

The SHA-256 algorithm is significant because it is used in Bitcoin mining as well as several other Proof of Work blockchain networks. Its importance is supported by the fact that Bitcoin (BTC) was the world's first digital currency and is still the most valuable virtual currency by market capitalization. The SHA-256 algorithm is important since it is used in Bitcoin mining.

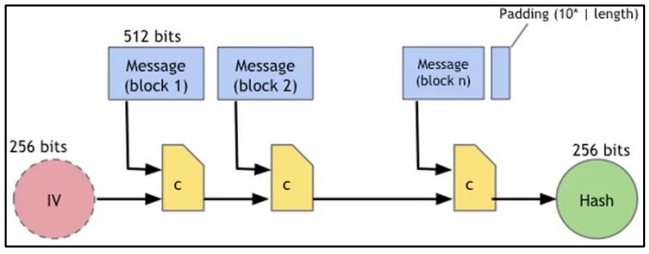

Fig 1. SHA-256 Algorithm.

In addition, SHA-256 has quite good technical parameters [4]:

- Block size indicator (byte): 64.
- Maximum allowed message length (bytes): 33.
- Characteristics of the message digest size (bytes): 32.
- Standard word size (bytes): 4.
- Internal position length parameter (bytes): 32.
- Number of iterations in one cycle: 64.
- Speed achieved by the Protocol (MiB/s): approximately 140.

4. Proposed System

Figure 2 shows the block diagram of the proposed system for Certificate generation and verification system using Blockchain Technology.

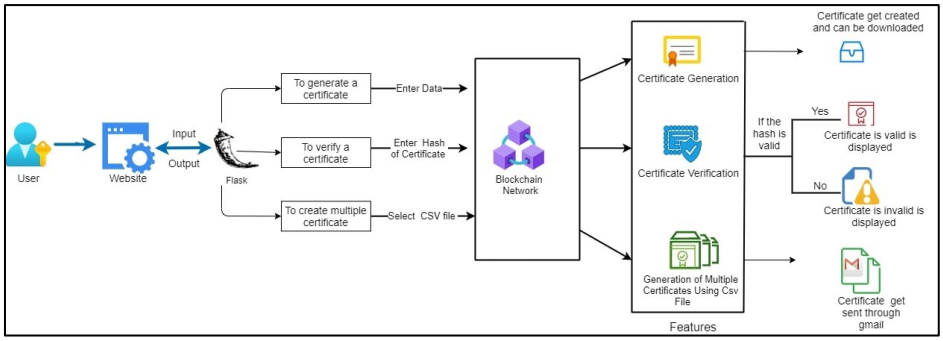

Fig 2. Block diagram of the proposed system.

An organization may communicate with the proposed system's user interface (UI) to access various services such as certificate issuance, verification, and retrieval in the above proposed system. Note that the proposed framework includes functionality such as certificate validation and generation. Flask is used to submit the user's input to the backend. Flask is a micro web framework scripted in Python that acts an arbiter among frontend and backend. Output is brought by backend utilizing Flask. User needs to register on the portal using email and password using this account user needs to login to the account and request the documents needed.

Our system is basically divided into three main stages:

- **Issue the documents:** Once the issuer verifies the credentials from the user, the issuer generates the certificate with unique and secured hash and put it on a secured blockchain network. The user can download a digital copy of the certificate after it has been created.
- **Blockchain:** It is a secured network in which documents are stored using hash which are connected in blocks through hash.
- **Retrieval of document:** User can view the document using the unique hash provided. Users can provide the hash to others to verify the document. These hash can be obtained by everyone due to the characteristics of the Blockchain. As a result, anybody with access to the Blockchain can now confirm the legitimacy of a certificate without relying on third parties.

The processing of output pertaining to input is carried out using flask in association with blockchain and SHA-256 Algorithm.

5. Implementation

Proposed system have been implemented using following steps with the help of different blockchain functionalities and methods.

- Taking Input from user and Generating BlockHash
- Certificate Generation
- Certificate Verification
- Sending Certificates through email

A. *Taking Input from user and Generating BlockHash*

The hashlib module provides a standardized interface to a variety of stable hash and message digest algorithms. The SHA1, SHA224, SHA256, SHA384, and SHA512 stable hash algorithms as well as RSA's MD5 algorithm are included in this module [12]. The algorithm sha256 is used to produce blockhash from all of this.

```
class Block:
    def __init__(self, previous_block_hash, data, timestamp):
        self.previous_block_hash = previous_block_hash
        self.data = data
        self.timestamp = timestamp
        self.hash = self.getHash()

    @staticmethod
    def createGenesisBlock():
        return Block("0", "0", datetime.datetime.now())

    def getHash(self):
        header_bin = (str(self.previous_block_hash) +
                      str(self.data) +
                      str(self.timestamp)).encode()

        innerHash = hashlib.sha256(header_bin).hexdigest().encode()
        outerHash = hashlib.sha256(innerHash).hexdigest()
        return outerHash
```

Pseudo code 1: Generating BlockHash

Block class generates blockhash using previous block hash, data, and timestamp as input. If the code is executed for the first time, that is, if no prior blockhash exists, the createGenesisBlock( ) function is named, which creates the first blockhash with default values before generating the blockhash.The getHash( ) method returns the value after creating a blockhash using the Sha256 algorithm.

B. *Certificate Generation*

The GenerateCertificate( ) method is called to create certificate. This method takes name, blockhash and college name as input.

```
def GenerateCertificate(name, block, clg):
    font = ImageFont.truetype('arial.ttf', 80)
    font2 = ImageFont.truetype('arial.ttf', 50)
    font3 = ImageFont.truetype('arial.ttf', 40)

    img = Image.open(
        'C:/Users/asus/Desktop/Blockchain app/website/static/certificate.jpg')
    draw = ImageDraw.Draw(img)
    draw.text(xy=(675, 480), text='{}'.format(name), fill=(0, 0, 0), font=font)
    draw.text(xy=(450, 810), text='{}'.format(clg), fill=(0, 0, 0), font=font2)
    draw.text(xy=(450, 1180), text='{}'.format(
        block), fill=(0, 0, 0), font=font3)
    img.save(
        'C:/Users/asus/Desktop/Blockchain app/website/static/SaveCertificate/
        {}.jpg'.format(name))
```

Pseudo code 2: Certificate Generation

C. *Verify Certificate*

The verifyBlock( ) method is used to verify the certificate. This method accepts a blockhash as an input that must be checked. If blockhash is valid, the certificate data is shown to the user; if the certificate is invalid, the message “certificate is invalid” is shown.

```
def verifyBlock():
    blockHash = request.form.get("block")

    blockCheck = block_chain_data.query.filter_by(blockHash=blockHash).first()
    if blockCheck:
        data={"Status":"Certificate Valid" ,"Block Hash": blockHash,
              "Name": blockCheck.fullName, "College Name": blockCheck.clgName}
      return render_template("verifiedDetails.html", data=data)
    else:
      data={"Status":"Certificate Invalid"}
      return render_template("verifiedDetails.html", data=data)
```

Pseudo code 3: Certificate Verification

D. *Sending Certificates through email*

Certificates are automatically sent to the user by email when several certificates are generated. The smtplib module is used to send emails. SMTP is a network session object that can be used to transmit email to any system on the Internet that has SMTP or ESMTP listener daemon. This approach requires the input of an email address and a file name. The issuer’s workload is minimized by sending certificates via email.

```
def sendEmail(email, filename):

    fromaddr = "senderEmail"
    toaddr = email
    msg = MIMEMultipart()

    msg['From'] = fromaddr
    msg['To'] = toaddr
    msg['Subject'] = "BlockSign Certificate"


    msg.attach(MIMEText(body, 'plain'))

    filename = "file_name.jpg"
    attachment = open("path", "rb")

    p = MIMEBase('application', 'octet-stream')
    p.set_payload((attachment).read())
    encoders.encode_base64(p)
    p.add_header('Content-Disposition', "attachment; filename= %s" % filename)
    msg.attach(p)
    s = smtplib.SMTP('smtp.gmail.com', 587)
    s.starttls()
    s.login(fromaddr, "password")
    text = msg.as_string()
    s.sendmail(fromaddr, toaddr, text)
    s.quit()
```

Pseudo code 4: Certificate sending through email

6. Result Analysis

The home page appears after users log in to the system, as seen in Figure 3. The user has three options on the home page: create certificate, verify certificate, and CMC (create multiple certificate). The user can only create one certificate at a time using the create option. The user can verify the certificate by entering the blockhash in the verify option. The CMC alternative allows the user to generate different certificates from a CSV file. Any alternative can be selected by the user depending on their requirements.

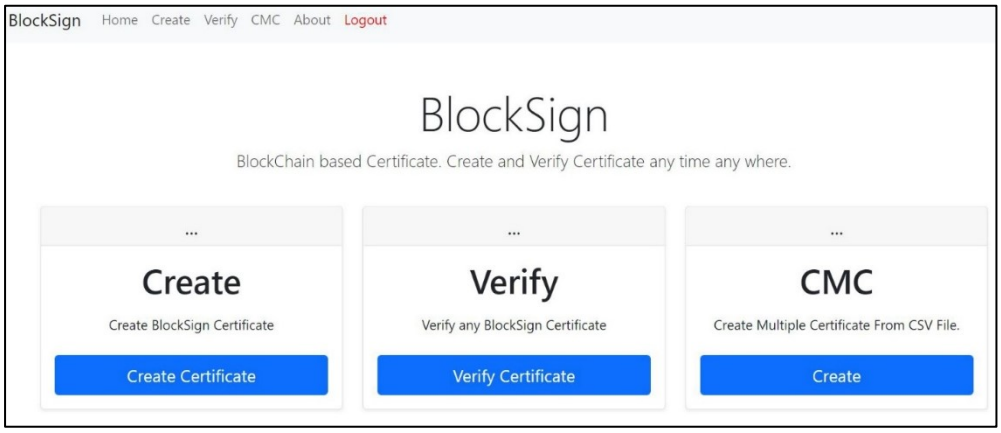

Fig. 3. Home Page.

Figure 4 shows the page that appears when the user selects the option to create a certificate. The user must enter their Full Name and College Name. The certificate is created after you press the submit button.

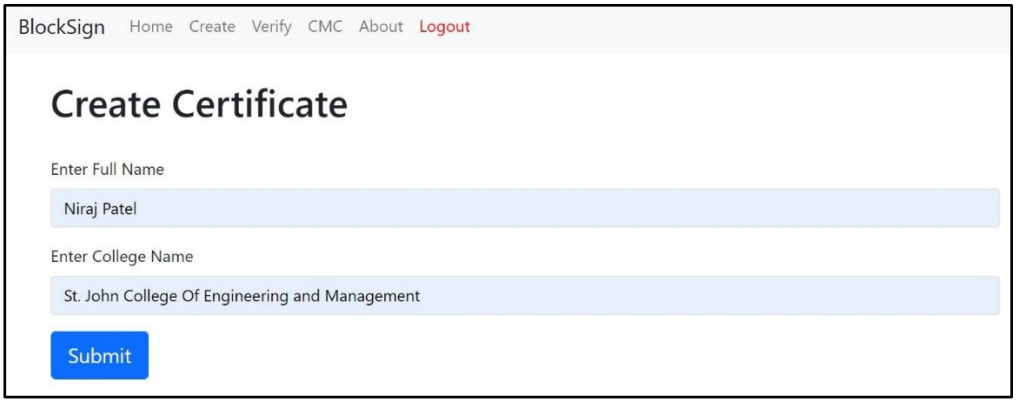

Fig. 4. Certificate Creation.

Figure 5 displays the page that appears when the user hits the submit button. The user can see the certificate information here, and the generated certificate can be downloaded by pressing the download certificate button.

Fig. 5. Certificate Details.

Figure 6 shows the certificate that is produced. The credential includes the user's information as well as the blockhash, which is the cryptographic signature. Anyone can validate the certificate using this blockhash.

Fig. 6. Certificate Generation.

If a user needs to validate a certificate, they must enter blockhash in the format shown in Figure 7. The credential details are shown if the blockhash is correct. If the blockhash is invalid, the message "Certificate Not Valid" will appear.

Fig. 7. Certificate Verification.

If the blockhash entered is right, the page depicted in figure 8 is shown. All of the certificate's details are shown here.

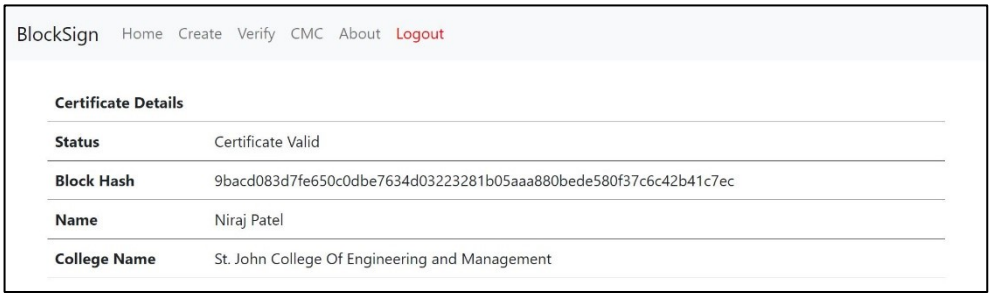

Fig. 8. Certificate Validation.

Figure 9 shows how a user can use a CSV file to create several certificates. The user must first pick the CSV file and then click the submit button. After that, the certificate will be produced and sent to the user by email, with all of the relevant information displayed.

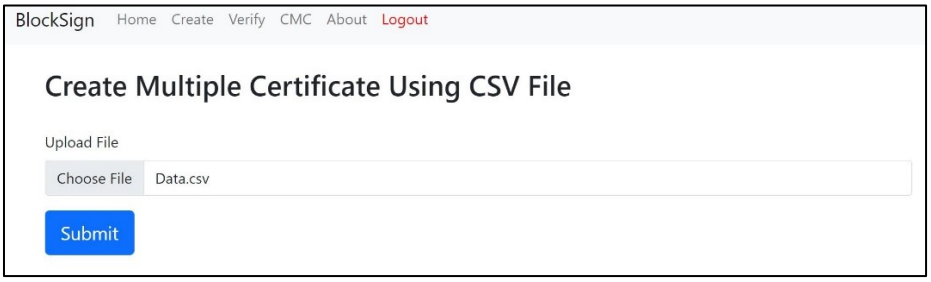

Fig. 9. Creation of multiple Certificates.

The CSV file with data and email is shown in Figure 10. This file's data will be used to create a certificate, and the email address in this file will be used to send an email to the recipient. The file will be attached to the email, which will then be forwarded to the recipient.

	A	B	C
1	name	clg_name	email
2	Niraj Patel	St. John College of Engineering and Management	pniraj657@gmail.com
3	Ranjan Patel	St. John College of Engineering and Management	nirajp@sjcem.edu.in
4	Sukaji Parab	St. John College of Engineering and Management	sukajip@sjcem.edu.in
5	Sushil Maurya	St. John College of Engineering and Management	sushilm@sjcem.edu.in
6	Mathews Joel	St. John College of Engineering and Management	joelgmathews@gmail.com
7			

Fig. 10. CSV File.

The email is sent to the user on the email id given in the CSV file as seen in Figure 11. This email contains the user's certificate.

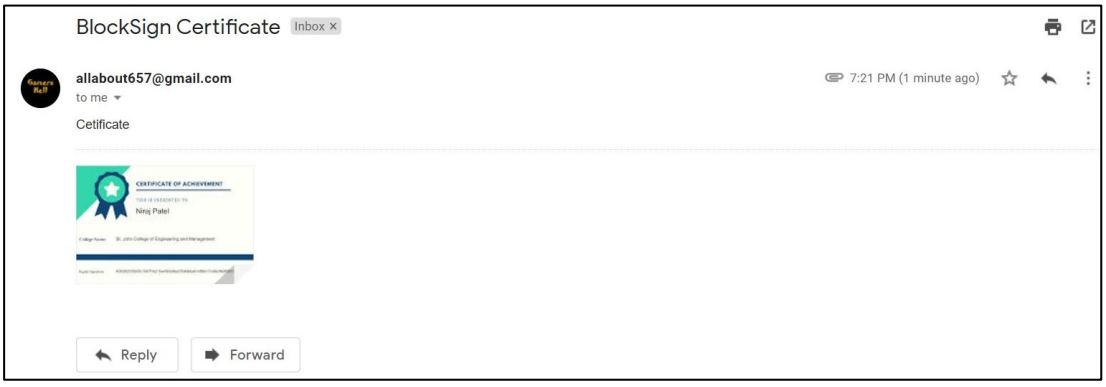

Fig. 11. Email received with certificate

7. Conclusion and Future Scope

The aim for this project is to create tamper proof certificates using Blockchain. The environmental effect of blockchain is also very positive because it is an entirely automated method that eliminates the need for paper. The user's workload is also minimised since this system allows them to automatically generate and send certificates via email. The authentication process has also been shown to have a low failure rate, meaning that people would not need to contact the issuing authority several times if blockchain technology is used. Using this immutable approach, this system can be expanded to generate secure identification documents such as Aadhar cards, PAN cards, passports, and so on.

REFERENCES

[1] C. Jeena, How students and employers can spot and eliminate fake degrees, India Today, Sept 27 2020. [Online]. Available: https://www.indiatoday.in/education-today/featurephilia/story/how- students-and-employers-can-spot-and-eliminate-fake-degrees-1725931- 2020-09-27[Accessed on November 2020].

[2] "Human Rights Watch," 2 February 2011. [Online]. Available: https://www.hrw.org/report/2011/02/01/anti-nationals/arbitrary- detention-and-torture-terrorism-suspects-india. [Accessed on November 2020].

[3] "Splunk," [Online]. Available: https://www.splunk.com/en-us/data- insider/what-is-blockchain.html. [Accessed on December 2020].

[4] S. Patel and S. Shah, "High Performance Cryptographic Hash Function On FPGA Using SHA-256," International Journal Of Scientific Progress and Research (IJSPR), vol. 34, no. 97, 2017.

[5] "Bitcoin wiki," [Online]. Available: https://en.bitcoinwiki.org/wiki/SHA-256. [Accessed on December 2020].

[6] BHOWMIK, Deepayan and FENG, "The multimedia blockchain: a distributed and tamper-proof," in 22nd international conference, London, 2017.

[7] M. shah and p. Kumar, "Tamper Proof Birth Certificate Using Blockchain Technology," International Journal of Recent Technology and Engineering, vol. 7, no. 583, 2019.

[8] N. Kumavat, S. Mengade, D. Desai and J. varolia, "Certificate Verification System using Blockchain," International Journal for Research in Applied Science and Engineering Technology (IJRASET), vol. 7, no. 4, 2019.

[9] G. Malik, K. Parasrampuria, S. P. Reddy and S. Shah, "Blockchain Based Identity Verification Model," in International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN), 2019.

[10] P. Kumar, K. K. Kumar, R. s. Krishna and P. A. Shri, "Incorporation of Blockchain in Student management System," International Journal of Innovative Technology and Exploring Engineering (IJITEE), vol. 8, no. 6, 2019.

[11] P. Vadhera and B. Lall, "Review Paper on Secure Hashing Algorithm and Its Variants," International Journal of science and Research(IJSR), vol. 3, no. 6, pp. 629-632, 2014.

[12] Python library-Hashlib, "Hashlib," [Online]. Available: https://pypi.org/project/hashlib/. [Accessed November 2020].

[13] G. Capece, N. L. Ghiron and F. Pasquale, "Blockchain technology: Redifining trust for digital certificates," Sustainibility, 2020.

[14] G. Shankar, A. Dravid, K. M and D. B, "Blockchain based Certificate Issuing and Validation," International Research Journal of Engineering and Technology (IRJET), vol. 06, no. 03, 2019.

[15] Python library-Crypto, "crypto," [Online]. Available: https://pypi.org/project/crypto/. [Accessed on November 2020].

[16] Python library-Flask, "Flask," [Online]. Available: https://pypi.org/project/Flask/. [Accessed on December 2020].