Integration of Academics in Blockchain

R. Brindha, Om Trikha and Prithvi Velsa

Abstract— *Blockchain is a distributed record keeping system. It stores records in the form of encrypted blocks that are linked or "chained" together. Each block has its own generated hash and the hash of the previous block which allows for greater security. It creates a decentralized environment where the stored records as well as all the transactions concerning the records is not controlled by any one single entity but rather all of them. Any completed transaction is stored in these blocks in a permanent, immutable and completely verifiable and transparent way. A sector where Blockchain can be applied for great benefits is academics. Academics are defined as the authorities tasked with the task of imparting education in an institution. Blockchain is a rapidly growing technology that can definitely revolutionize some aspects related to academia. This paper will review some such applications as well as an implementation of a private blockchain purpose built for a University's grade entering system. This blockchain will be fully functional and offer all of the facilities of a merkle tree which is the standard for any application.*

Keywords--- *Blockchain - Academia - Data Security - Distributed Ledger - Decentralized Server*

I. INTRODUCTION

Data Security is becoming more and more vital in our modern world. Any records with real world impact need to be stored in a secure manner such that no illegitimate updations take place. One such sector is academics. As of now, a majority of educational institutions save their data of student grades, credits, attendance and other such parameters on traditional central server based databases. While these are encrypted and reasonably secured behind layers of defense, people are getting more and more competent at cracking through these firewalls. It would be very disastrous if unauthorized persons make changes to a record leading to unfair evaluations and all the ramifications that occur as a consequence. Therefore, storing data in a blockchain alleviates these security vulnerabilities.

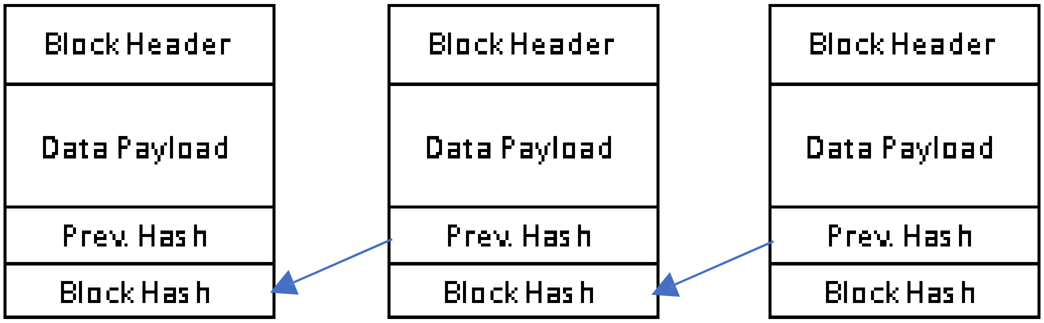
A. Blockchain Overview

A Blockchain in short, is a decentralized, distributed ledger. It is a chronologically sequential and immutable (meaning it can't be changed) chain of records called blocks that are linked together by hash values. Since it is decentralized, there is no central server maintaining the data – instead, every node on the blockchain network has its copy of the blockchain stored that is synced via the network periodically. Blockchain network is connected peer to peer, meaning that each node on the system is connected to the other nodes itself, instead of having a centralized connection point. Each block in the blockchain can contain data, files, transactions or any other payload as needed and the entire thing is cryptographically encrypted by using an algorithm (most popularly SHA-26). It will also contain a hash value of its contents and another hash value that is a reference to the previous block in the

R. Brindha, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Kattankulathur, Tamilnadu.
Om Trikha, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Kattankulathur, Tamilnadu.
Prithvi Velsa, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Kattankulathur, Tamilnadu.

DOI: 10.37200/IJPR/V24I5/PR201926
Received: 20 Feb 2020 | Revised: 14 Mar 2020 | Accepted: 28 Mar 2020

sequence. The first block in the blockchain is also called the nonce and it is the only block that will not have the previous hash value field filled. The chain is formed by finding the next block whose previous hash value matches the current block that is linked. Since it is impossible to ensure real time synchronization takes place flawlessly every single time, nodes on the blockchain network accept the longest valid chain as the authoritative chain.



For the creation of a new block, nodes will have to expend some time doing work (termed as 'mining') to generate a value that is easily verifiable in the blockchain but not easy to generate through running complex mathematical sequences. This mining process is accomplished by the usage of large amounts of computing power. This is termed as 'Proof of Work' and is a vital part in maintaining the integrity of the blockchain.

Blockchain is used in a wide range of sectors from banking to business governance to healthcare. While there is a certain amount of inertia in adoption of any new technology, blockchain has found it's niche in some areas – particularly in the realm of cryptocurrencies. It can even be regarded as the backbone of the entire sector. Arguably the most famous cryptocurrency, Bitcoin catapulted the blockchain technology into the limelight. Developed by the enigmatic personality Satoshi Nakamoto, this cryptocurrency slowly rose to prominence by trading at exponentially high values and redefined international trading as well as spurring the growth of several new cryptocurrencies, none of which managed to reach the levels of success enjoyed by Bitcoin. There it is very evident that blockchain technology is tried and tested and is not just a temporary stopgap in the relentless march of technological solutions.

B. Blockchain and Academics

Blockchain offers a number of advantages that make it ideal for usage in academics. The easiest and primary purpose for which a blockchain can be used is to replace the traditional repository provided by enterprise databases. There are also other use cases for a blockchain that are more specialized such as using blockchain to verify and transact credit transfers in universities.[1][6] This is all possible through the unique benefits provided by the blockchain which are elaborated below.

A Blockchain is very transparent; meaning that every single transaction or addition is publicly viewable. Each block is time stamped and contains identification so that there is no ambiguity and accountability is maintained at all times which is vital in an academic environment. All stakeholders, namely teachers, administration and students can view the blockchain and assure themselves that the data is untampered with and secure. [4]. All blocks are immutable meaning that once it has been added to the blockchain, it cannot be changed or modified

DOI: 10.37200/IJPR/V24I5/PR201926
Received: 20 Feb 2020 | Revised: 14 Mar 2020 | Accepted: 28 Mar 2020

in any way. If a mistake has been made, it needs to be rectified in another block and care must be taken to ensure that the updation is legitimate. Having a record (a ‘paper trail’ of sorts) of all transactions due to the transparent nature of the blockchain makes it more secure. The fact that it is distributed on several nodes joined in peer to peer networks as well as the cryptographic encryption of the data within each block also makes it immune to any security breaches. [7]

Since the blockchain is a peer to peer sharing network, there is no centralized server. This cuts out the ‘middleman’ in each transaction thereby making every transaction faster, cheaper and more efficient in terms of transactional and logistical overhead. In an academic institution with potentially huge amounts of transactions every day, these savings will add up pretty quickly to present a significant benefit over a period of time.[2]

Another benefit of the blockchain being a peer to peer sharing network is that not having a centralized server translates into a much more secure and reliable framework. [3] In the traditional central server based networks, if the single main point (i.e. the central server) goes down due to issues or even some maintenance, the entire network goes down as well. In a blockchain, since every node is virtually its own entity with its own copy of the ledger, even if one or more nodes go down, the network isn’t affected. It also helps in the security aspect by denying hackers focusing all their efforts on a single ‘chink in the armor’.

II. State Of The Art

A. *“EduCTX: A blockchain based Higher Education Credit Platform” - Muhamed Turkanović, Marko Holbl, Kristjan Kosic, Marjan Hericko, Aida Kamisalic (5 Jan 2018)*

Proposes a barebones ‘EduCTX’ blockchain that is the basis of the EduCTX initiative, that is used for efficient transfer of credits in a simple, efficient and global way while minimizing language barriers and administrative obstacles. Introduces the idea that blockchains can be very useful in academics. [1]

B. *“The Blockchain Phenomenon – The Disruptive Potential of Distributed Consensus Architectures” - Mattila and Juri (10 May 2016)*

Talks about how consensus architecture or the need for multiple signatory authorities to verify a transaction in blockchains makes it all the more secure. Provides the idea of distributed ledger keeping. [2]

C. *“Blockchain: Future of financial and cyber security” - Sachchidanand Singh, Nirmala Singh (17 May 2017)*

Explains the different encryption standards used worldwide. Elaborates greatly on the security benefits of using a properly secured blockchain. [3]

D. *“Bitcoin Message: Data Insertion on a Proof-of- Work Cryptocurrency Systems” - Matthew D. Sleiman, Adrian, P. Lauf, Roman Yampolskiy (7 Oct 2015)*

Provides an overview of how the bitcoin blockchain works. Also goes into detail about how to insert data in blocks mined by miners for a Proof-of-Work concept. [11]

E. *“Federal Information Processing Standards Publication: secure hash standard,”- Q. H. Dang, (5 Aug 2015)*

Expounds on the importance of hashing standards and how they matter in terms of security. Advocates for the use of SHA-256 hashing algorithm. [9]

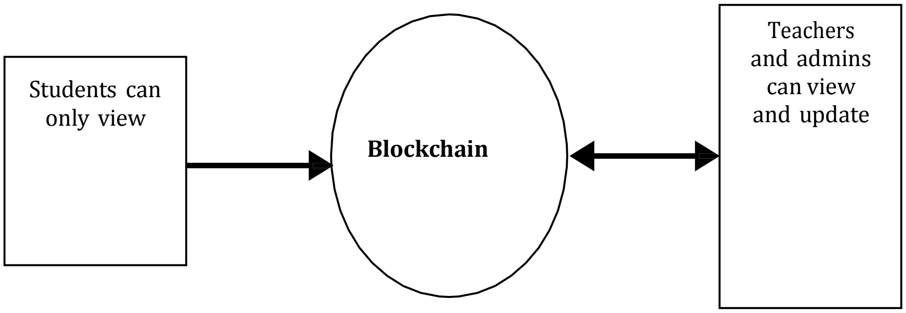
DOI: 10.37200/IJPR/V24I5/PR201926
Received: 20 Feb 2020 | Revised: 14 Mar 2020 | Accepted: 28 Mar 2020

III. Proposed Work

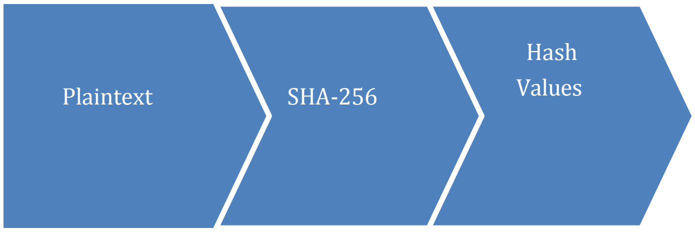
To provide real world data on the benefits of switching from traditional repositories to blockchain based record keeping, one of the more efficient applications that can be implemented is storing a University's academic data such as student grades for particular courses under different teachers in a blockchain. This renders the data immutable and extremely transparent which helps in maintaining the security of the data. [10]

The blockchain will have to be in two modules - one is the actual blockchain itself and the second being the web portal from which all the stakeholders can access the data in the blockchain from. The blockchain will interface directly with the web portal and any additions to the data will update in real time. The web portal should have different forms that branch out from a central authorization checkpoint (log-in page) and depending on the authority level of the stakeholder accessing it, will have differing levels of options. A stakeholder must be able to at the very least, view the data in chain with the higher levels in the hierarchy (i.e. teachers and administrators) must be able to add data to the blockchain as well. [5]

All possible safeguards and general good coding practices need to be followed to create a cohesive, functional and useful blockchain. Ease of use must be kept in mind and all the stakeholders need to derive tangible benefits from using it for their purposes.



The SHA-256 hashing algorithm standard will be used to generate the authentication hashes that are stored in each block. As previously explained, these hashes are vital to maintain the security of the blockchain as well as help in finding the next legitimate block in the current chain. SHA or Secure Hash Algorithm is an algorithm designed to create these cryptographic hashes. It is computed using 32 bit words and starts computing the hash for the given input from a pre- determined unknown starting value. This ensures that the hash values are not able to be cracked by any outside party and the data contained within each block is properly secured.



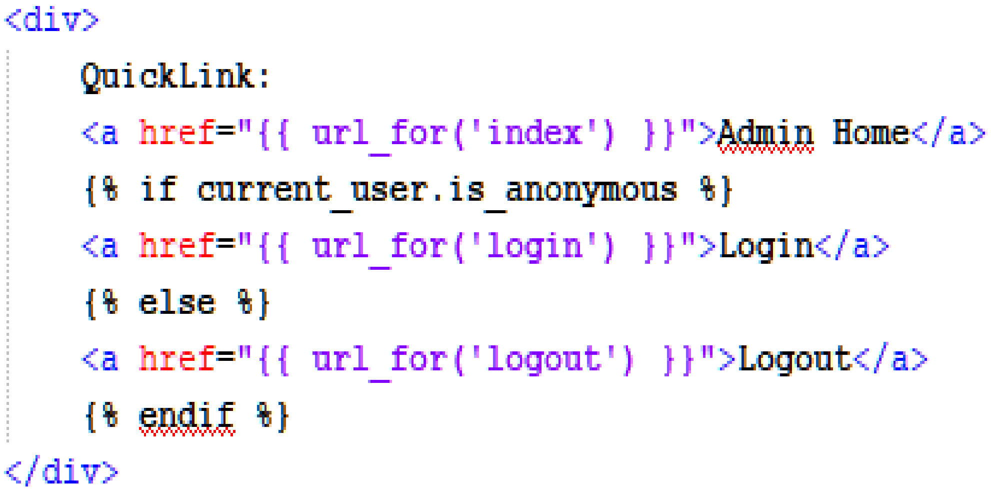
DOI: 10.37200/IJPR/V24I5/PR201926
Received: 20 Feb 2020 | Revised: 14 Mar 2020 | Accepted: 28 Mar 2020

For this particular use case – where all the stakeholders are linked together and there is some degree of trust, a completely private blockchain will be used. This is achieved by having the nodes connected to each other and the particular chain being accessible only to the nodes that are connected in that particular network; regular people i.e. people outside the network, will not be able to contribute to it.

Due to security concerns and the nature of a blockchain, false or incorrect data if entered, cannot be deleted – the updation is permanent and stays recorded as a transaction forever. It is up to the person to rectify their mistake in a future transaction and add a verifiable way to validate if whether the latest transaction is legitimate or not. One way to accomplish this is to use another verification system, manual or otherwise and adding a reference to that in a text field of the box.

IV. Implementation

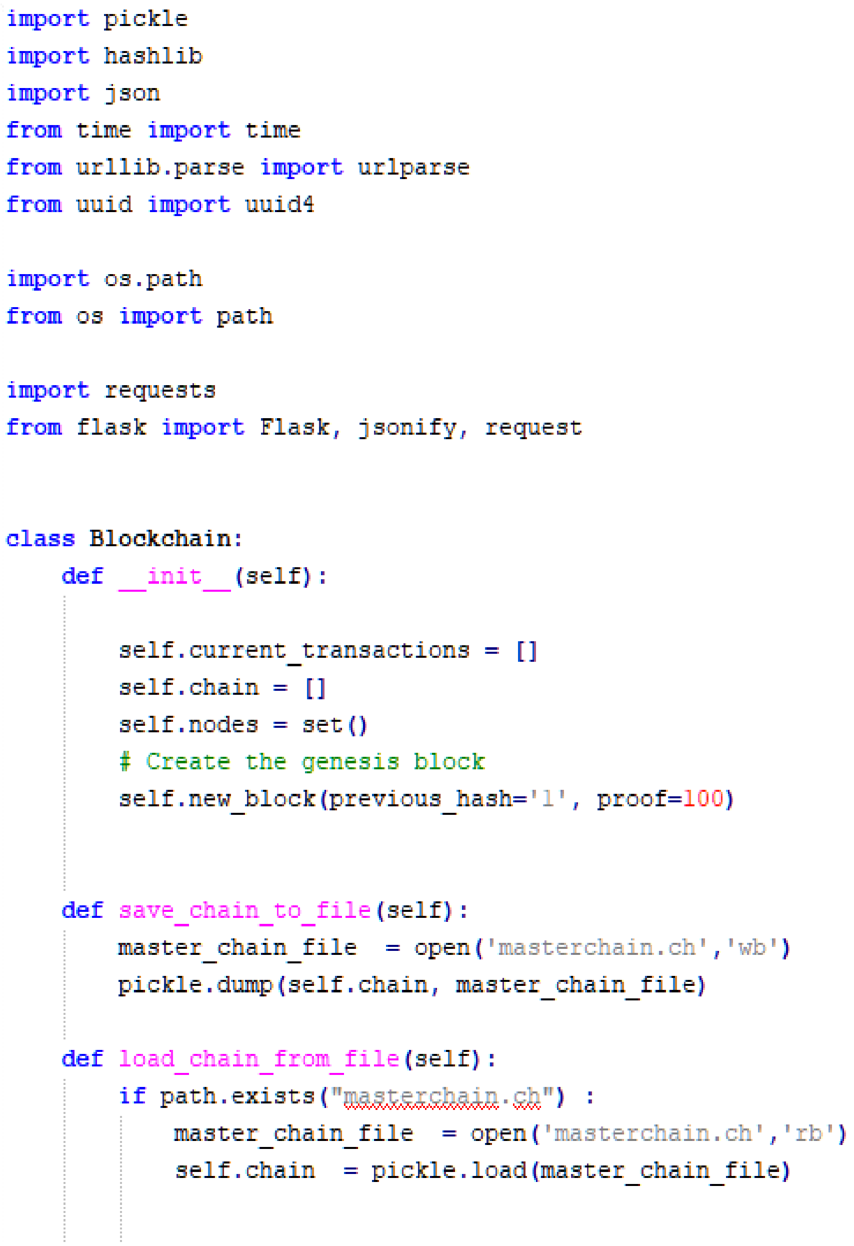
While the Blockchain itself can be programmed in many different languages, for the particular trial Python was chosen for its ease of usage and versatile network interfacing options. For the web server we will use Flask, a micro web framework. Flask will handle all the incoming and outgoing HTTP requests (which will be in the form of GET and POST). Flask will also take care of managing the SQLlite tables that will be used to store non-sensitive data such as course details and teacher details. The forms used to collect and display data will use Ginger templates so that the required information can be auto populated and formatted into an easily readable form.



```
<div>
    QuickLink:
    <a href="{{ url_for('index') }}">Admin Home</a>
    {% if current_user.is_anonymous %}
    <a href="{{ url_for('login') }}">Login</a>
    {% else %}
    <a href="{{ url_for('logout') }}">Logout</a>
    {% endif %}
</div>
```

The Blockchain will use the SHA256 hashing algorithm to encrypt the data. The python framework will use the pickle, JSON, uuid and hashlib libraries for all the different functions viz. creating a new block, authenticating a new block, replicating the blockchain on different nodes etc. The program will transfer the data in the JSON format after it translates from the encrypted hash block. Using the filter settings provided by the Flask library, the authentication level of the user currently logged in can be checked and only the relevant information shown to them/

DOI: 10.37200/IJPR/V24I5/PR201926
Received: 20 Feb 2020 | Revised: 14 Mar 2020 | Accepted: 28 Mar 2020

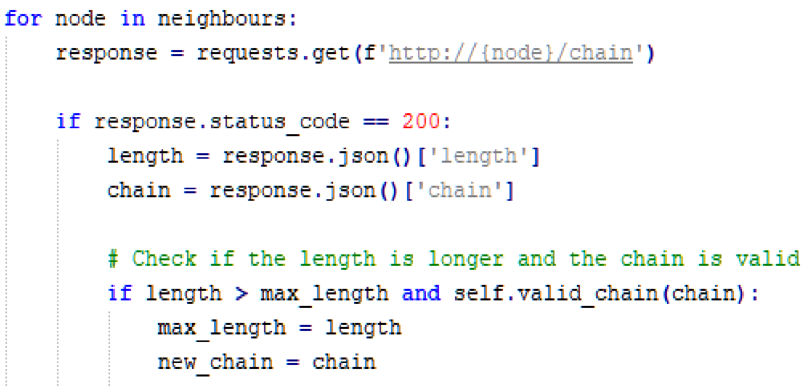


```
import pickle
import hashlib
import json
from time import time
from urllib.parse import urlparse
from uuid import uuid4

import os.path
from os import path

import requests
from flask import Flask, jsonify, request


class Blockchain:
    def __init__(self):

        self.current_transactions = []
        self.chain = []
        self.nodes = set()
        # Create the genesis block
        self.new_block(previous_hash='1', proof=100)


    def save_chain_to_file(self):
        master_chain_file  = open('masterchain.ch','wb')
        pickle.dump(self.chain, master_chain_file)

    def load_chain_from_file(self):
        if path.exists("masterchain.ch") :
            master_chain_file  = open('masterchain.ch','rb')
            self.chain  = pickle.load(master_chain_file)
```

He Blockchain itself will use the SHA256 hashing algorithm to encrypt the data. The python framework will use the pickle, JSON, uuid and hashlib libraries for all the different functions viz. creating a new block, authenticating a new block, replicating the blockchain on different nodes etc. The program will transfer the data in the JSON format after it translates from the encrypted hash block. Using the filter settings provided by the Flask library, the authentication level of the user currently logged in can be checked and only the relevant information shown to them.

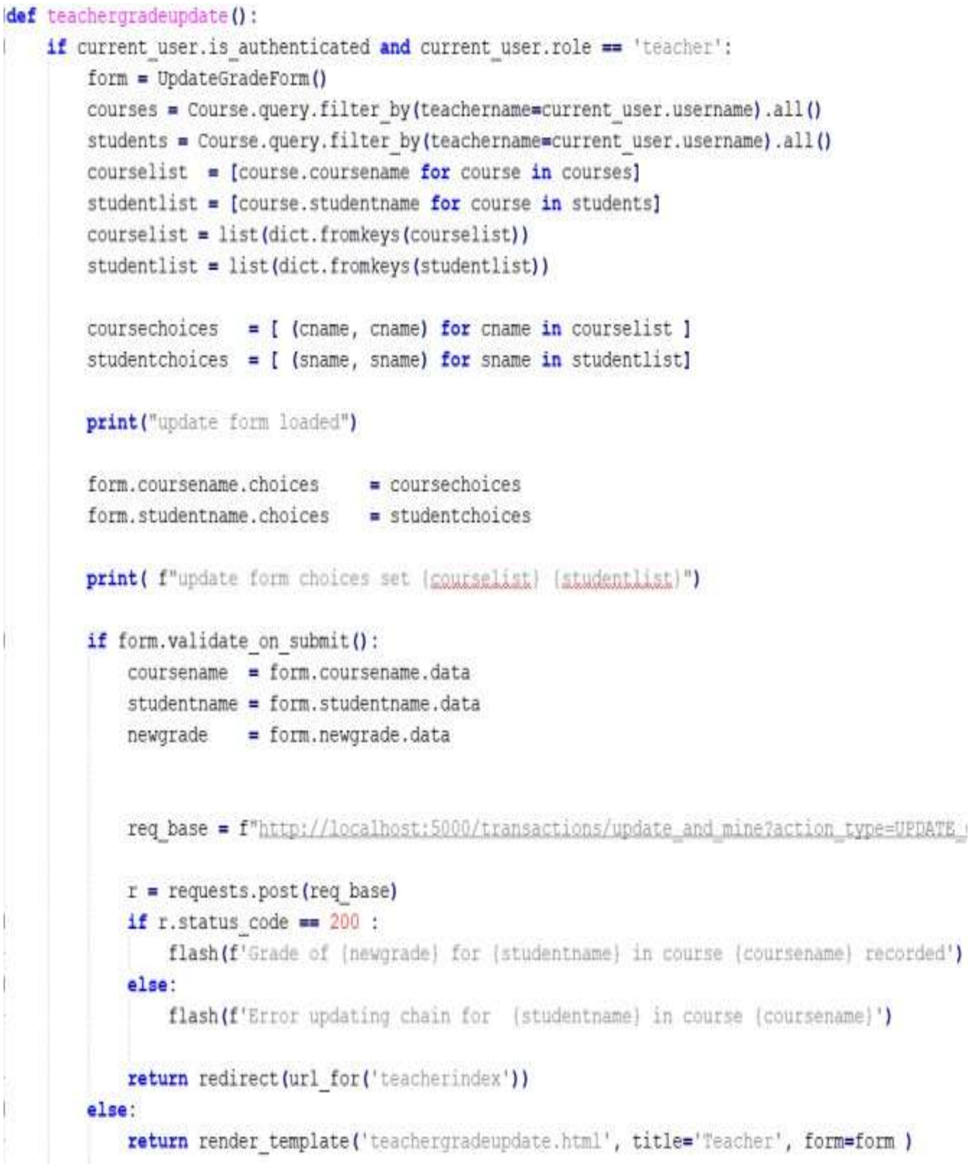
For the consensus algorithm, the “longest valid chain” will be used, which essentially means that the longest chain containing the most number of valid blocks with matching hash numbers will be chosen as the legitimate chain and all the other nodes will sync to that particular chain. [8]

DOI: 10.37200/IJPR/V24I5/PR201926
Received: 20 Feb 2020 | Revised: 14 Mar 2020 | Accepted: 28 Mar 2020



```
for node in neighbours:
    response = requests.get(f'http://{node}/chain')

    if response.status_code == 200:
        length = response.json()['length']
        chain = response.json()['chain']

        # Check if the length is longer and the chain is valid
        if length > max_length and self.valid_chain(chain):
            max_length = length
            new_chain = chain
```

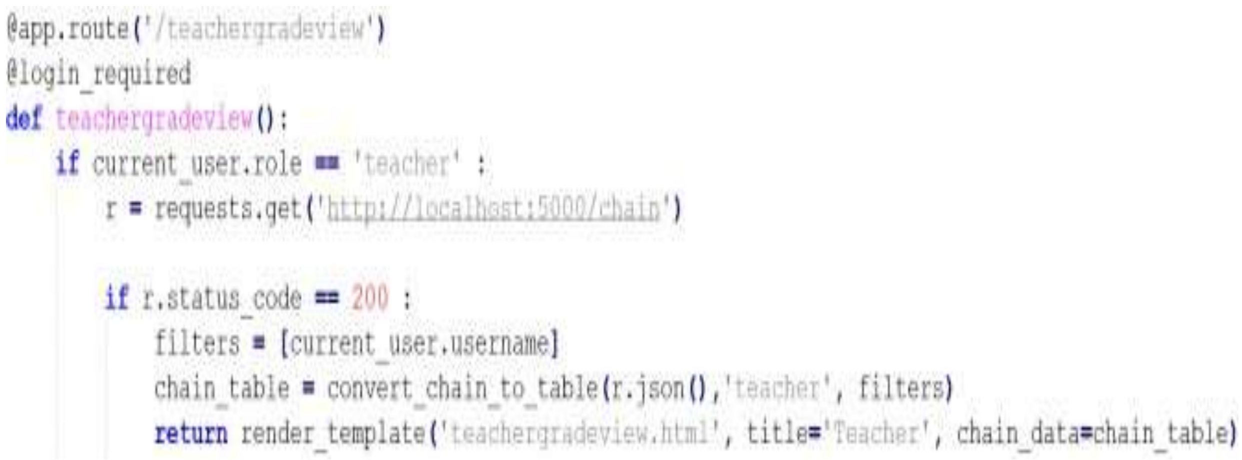
For teachers, to view the specific student details, we will need to apply some filters on the data that is being interfaced from the database, so that only the students that belong to the courses they (the teachers) are assigned to are shown.



```
def teachergradeupdate():
    if current_user.is_authenticated and current_user.role == 'teacher':
        form = UpdateGradeForm()
        courses = Course.query.filter_by(teachername=current_user.username).all()
        students = Course.query.filter_by(teachername=current_user.username).all()
        courselist  = [course.coursename for course in courses]
        studentlist = [course.studentname for course in students]
        courselist = list(dict.fromkeys(courselist))
        studentlist = list(dict.fromkeys(studentlist))

        coursechoices   = [ (cname, cname) for cname in courselist ]
        studentchoices  = [ (sname, sname) for sname in studentlist]

        print("update form loaded")

        form.coursename.choices     = coursechoices
        form.studentname.choices    = studentchoices

        print( f"update form choices set {courselist} {studentlist}")

        if form.validate_on_submit():
            coursename  = form.coursename.data
            studentname = form.studentname.data
            newgrade    = form.newgrade.data


            req_base = f"http://localhost:5000/transactions/update_and_mine?action_type=UPDATE_

            r = requests.post(req_base)
            if r.status_code == 200 :
                flash(f'Grade of {newgrade} for {studentname} in course {coursename} recorded')
            else:
                flash(f'Error updating chain for  {studentname} in course {coursename}')

            return redirect(url_for('teacherindex'))
        else:
            return render_template('teachergradeupdate.html', title='Teacher', form=form )
```

DOI: 10.37200/IJPR/V24I5/PR201926
Received: 20 Feb 2020 | Revised: 14 Mar 2020 | Accepted: 28 Mar 2020

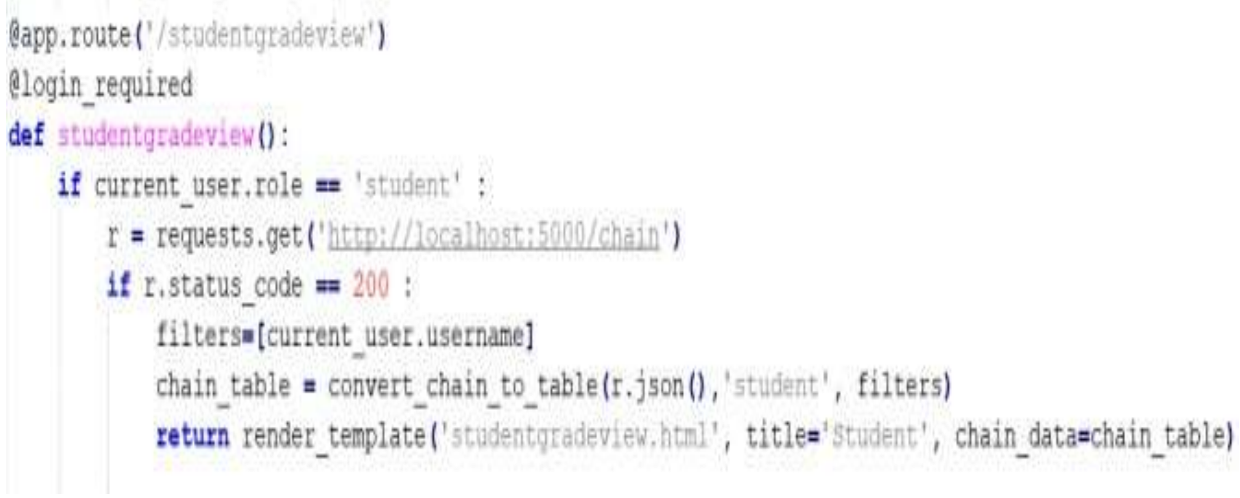
It needs to be noted that, the Flask form needs to double check the authorization level of the person currently logged in to make sure that whatever operation is being carried out is actually permissible for their access level.



```
@app.route('/teachergradeview')
@login_required
def teachergradeview():
    if current_user.role == 'teacher' :
        r = requests.get('http://localhost:5000/chain')

        if r.status_code == 200 :
            filters = [current_user.username]
            chain_table = convert_chain_to_table(r.json(),'teacher', filters)
            return render_template('teachergradeview.html', title='Teacher', chain_data=chain_table)
```

Similarly, for students, filters need to be applied to clearly indicate only the subjects they have registered for and show them their grades in those subjects.

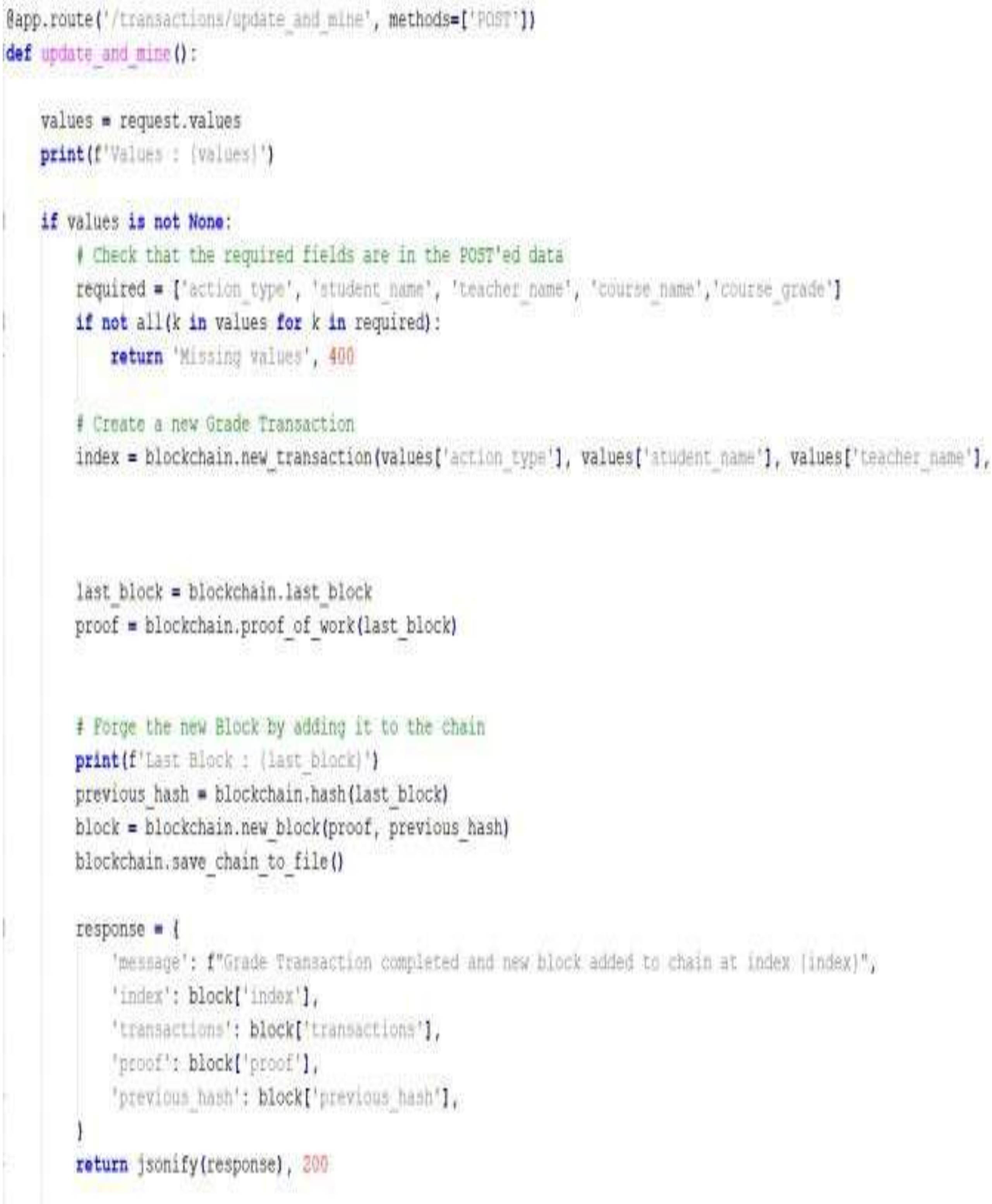
They also must not be able to see anyone else's grades, even if they share same parameters like same course name or same professor.



```
@app.route('/studentgradeview')
@login_required
def studentgradeview():
    if current_user.role == 'student' :
        r = requests.get('http://localhost:5000/chain')
        if r.status_code == 200 :
            filters=[current_user.username]
            chain_table = convert_chain_to_table(r.json(),'student', filters)
            return render_template('studentgradeview.html', title='Student', chain_data=chain_table)
```

For adding transactions to the chain, the nodes will have to 'mine' and add all the previous transactions, the payload (the grades) and the required hash values.

DOI: 10.37200/IJPR/V24I5/PR201926
Received: 20 Feb 2020 | Revised: 14 Mar 2020 | Accepted: 28 Mar 2020



```
@app.route('/transactions/update_and_mine', methods=['POST'])
def update_and_mine():

    values = request.values
    print(f'Values : {values}')

    if values is not None:
        # Check that the required fields are in the POST'ed data
        required = ['action_type', 'student_name', 'teacher_name', 'course_name','course_grade']
        if not all(k in values for k in required):
            return 'Missing values', 400

        # Create a new Grade Transaction
        index = blockchain.new_transaction(values['action_type'], values['student_name'], values['teacher_name'],



        last_block = blockchain.last_block
        proof = blockchain.proof_of_work(last_block)


        # Forge the new Block by adding it to the chain
        print(f'Last Block : {last_block}')
        previous_hash = blockchain.hash(last_block)
        block = blockchain.new_block(proof, previous_hash)
        blockchain.save_chain_to_file()

        response = {
            'message': f"Grade Transaction completed and new block added to chain at index {index}",
            'index': block['index'],
            'transactions': block['transactions'],
            'proof': block['proof'],
            'previous_hash': block['previous_hash'],
        }
        return jsonify(response), 200
```

All of this combine to form a private blockchain that is set up and running between different nodes on the university’s network. Teachers can use these different nodes to upload the grades of the students and the students can view their marks and grades easily.

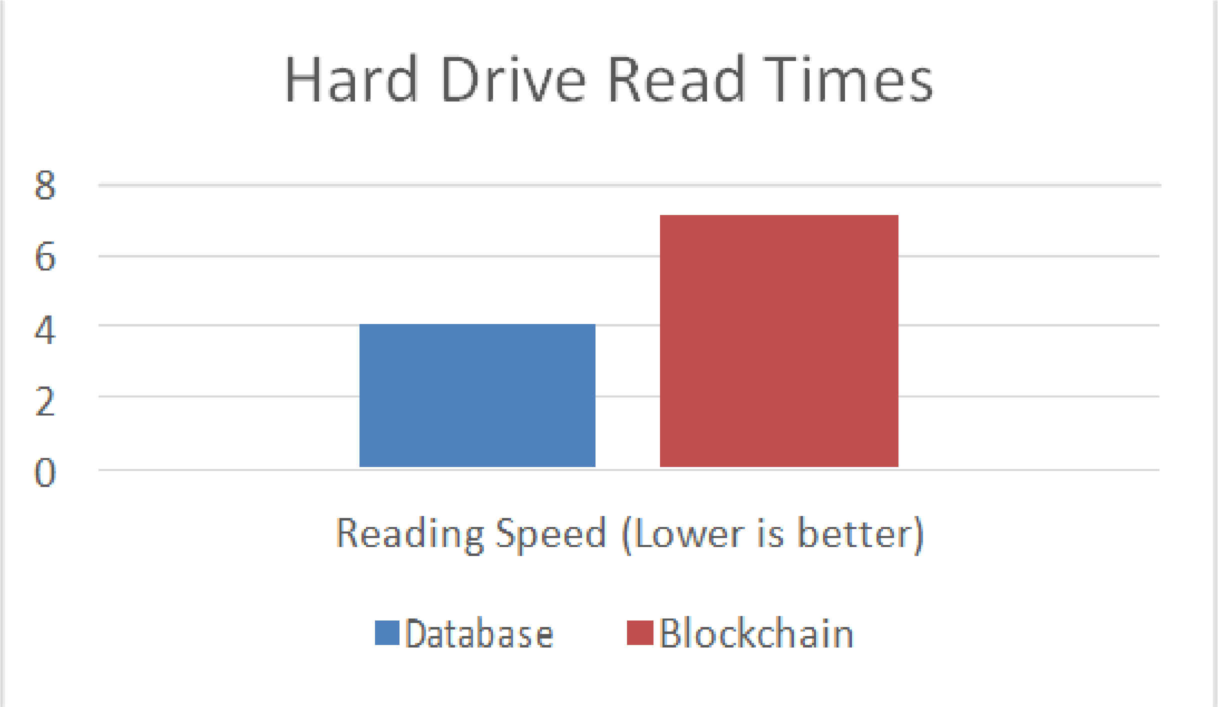
DOI: 10.37200/IJPR/V24I5/PR201926
Received: 20 Feb 2020 | Revised: 14 Mar 2020 | Accepted: 28 Mar 2020

V. RESULT DISCUSSION

If we compare the proposed solution of using a blockchain for record keeping instead of a regular database, externally, the end user will face no difficulty in switching systems. While there might be slight differences in how one accesses and adds new data, it should functionally be the same.

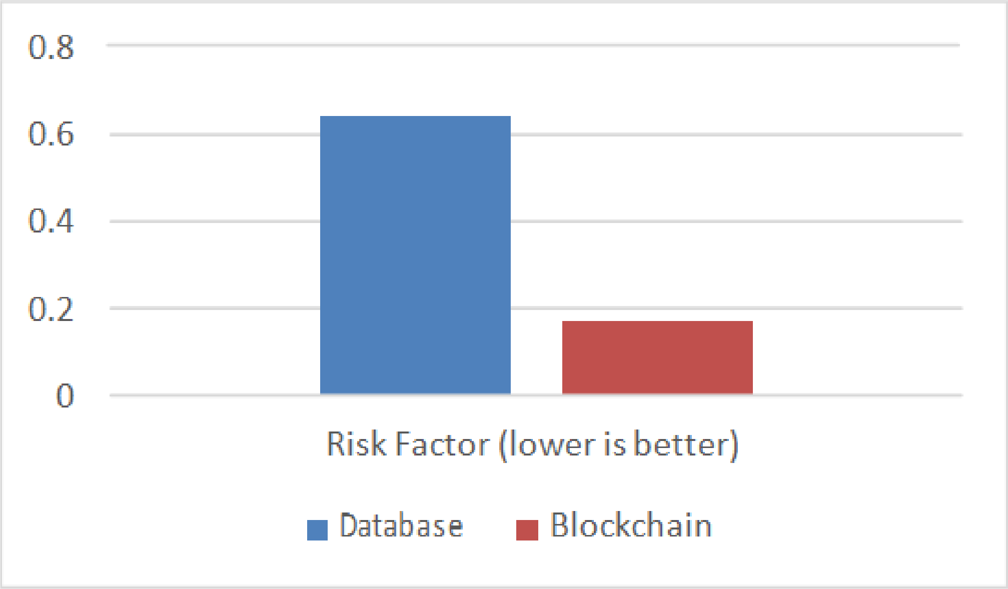
The main area of benefit that will be shown is in security. Blockchain, due to its inherently better ledger keeping system. Maintaining an immutable and transparent record of every transaction done makes falsification or illicit duplicity all but impossible. Having such a strong base in reliability makes the blockchain ideal for the proposed system of replacing the traditional repositories.

When both solutions are compared side by side, some clear differences between the two serve to contrast them both compared to each other. The two major categories where there are the most major differences is obviously security, and also speed.



When we compare the load times (the time taken to access the entirety of the data) for the same dataset consisting of 15 columns and 960 rows, the database implementation shows a distinct advantage in accessing the data. This is due to networks being able to run CRUD tests (Create Read Update Delete) much faster because all they have to do is authenticate themselves from one single central server while blockchain will need to perform several more operations like searching for other blocks and ensuring that the operation happens on the longest current chain. Therefore there is a serious speed (and space – blockchains store some data locally) issues with it.

DOI: 10.37200/IJPR/V24I5/PR201926
Received: 20 Feb 2020 | Revised: 14 Mar 2020 | Accepted: 28 Mar 2020



On the other hand, there is a clear winner when it comes to security. With risk factor being as the likelihood of a data breach in case of a determined and planned attack with ample resources from an opposing side, it is clear that the traditional database can lead to disaster much more easily than a blockchain. Which is why blockchain, even if slower, does not comprise on security thereby making sure any sensitive data remains immutable and secure even while being transparent. Which makes it an ideal fit for the academia scene as outlined above – since security is a higher priority than the speed of updations in most cases.

VI. Conclusion

Due to the very nature of the blockchain, using it over a traditional repository has many benefits. While the primary use case here is for the obvious security upgrade, the fact that it boasts features like seamless synchronization and transparency make it a better fit for storing records in certain cases. There are downsides such as consummation of space and being slightly slower but in today's world, storage space has become increasingly cheap which makes the proposition very enticing. Blockchains are the future and slowly lot of institutions, both academic and otherwise will start moving towards it and adopting it. Soon, there will be many sectors that depend on blockchain as a primary technology.

References

[1] M. Turkanovic, M. Holbl, K. Kosic, M. Hericko, and A. Kamisalic, "EduCTX: A Blockchain- Based Higher Education Credit Platform," *IEEE Access*, vol. 6, pp. 5112– 5127, 2018.

[2] Mattila and Juri, "The Blockchain Phenomenon - The Disruptive Potential of Distributed Consensus Architectures, by Mattila, Juri," *ETLA Working Papers*, 10- May-2016. [Online]

[3] Sachchidanand Singh, Nirmala Singh. "Blockchain: Future of financial and cyber security" *IEEE*, 17 May 2017

[4] L. Hughes, Y. K. Dwivedi, S. K. Misra, N. P. Rana, V. Raghavan, and V. Akella, "Blockchain research, practice and policy: Applications, benefits, limitations, emerging research themes and research agenda", *International Journal of Information Management*, vol. 49, pp. 114– 129, 2019.

[5] A.K. Jain, "Blockchain Goes to School," Mar. 2019.

[6] Abhishek Srivastava; Pronaya Bhattacharya; Arunendra Singh; Atul Mathur; Om Prakash; Rajeshkumar Pradhan, "A Distributed Credit Transfer Educational Framework based on Blockchain", *IEEE, 28 March*, 2017

DOI: 10.37200/IJPR/V24I5/PR201926
Received: 20 Feb 2020 | Revised: 14 Mar 2020 | Accepted: 28 Mar 2020

[7] L. Wang, X. Shen, J. Li, J. Shao, and Y. Yang, "Cryptographic primitives in blockchains," *Journal of Network and Computer Applications*, vol. 127, pp. 43– 58, 2019.
[8] L. Lee, "Alternative Signature Schemes- Blockchain At Berkeley-Medium," 12-Feb- 2019.
[9] Q. H. Dang, "Federal Information Processing Standards Publication: secure hash standard," Aug. 2015.
[10] Harry Halpin, Marta Piekarska "Introduction to Security and Privacy on the Blockchain", *IEEE,* April 2017.
[11] Matthew D. Sleiman, Adrian P. Lauf, Roman Yampolskiy ,"Bitcoin Message: Data Insertion on a Proof-of-Work Cryptocurrency Systems" - 7 Oct 2015

DOI: 10.37200/IJPR/V24I5/PR201926
Received: 20 Feb 2020 | Revised: 14 Mar 2020 | Accepted: 28 Mar 2020