**BLOCKCHAIN BASED SOLUTION FOR VERIFICATION**

**OF EDUCATIONAL CERTIFICATES**

A Capstone Project report submitted

in partial fulfillment of requirement for the award of degree

**BACHELOR OF TECHNOLOGY**

in

**SCHOOL OF COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE**

by

**D.THARUN (2103A51262)**

**G.PAVAN KUMAR (2103A51337)**

**T.RITHWIK (2103A51379)**

**B.VASU (2103A51193)**

Under the guidance of

**Mr. N.VENKATESH**

Assistant Professor, School of CS&AI.



SR University, Ananthsagar,Warangal,Telagnana-506371

**SR University**

Ananthasagar, Warangal.

**

**CERTIFICATE**

This is to certify that this project entitled **“BLOCKCHAIN BASED SOLUTION FOR VERIFICATION OF EDUCATIONAL CERTIFICATES**" is the bonafied work carried out by **D.THARUN, G.PAVAN KUMAR, T.RITHWIK, B.VASU** as a Capstone Project for the partial fulfillment to award the degree **BACHELOR OF TECHNOLOGY** in **School of Computer Science and Artificial Intelligence** during the academic year 2024-2025 under our guidance and Supervision.

**Mr. N.VENKATESH Dr. Shashikala**

Assistant Professor, Professor & Head,

SR University School of CS&AI, Anathasagar,Warangal SR University Ananthasagar, Warangal.

**Reviewer-1 Reviewer-2**

Name: Name:

Designation: Designation:

Signature: Signature:

**ACKNOWLEDGEMENT**

We owe an enormous debt of gratitude toour Capstone project guide **Mr. N.VENKATESH, Asst.Professor** as well as Head of the School of CS&AI , **Dr. M.Sheshikala, Professor** andDean of the School of CS&AI, **Dr.Indrajeet Gupta Professor** for guiding us from the beginning through the end of the Capstone Project with their intellectual advices and insightful suggestions. We truly value their consistent feedback on our progress, which was always constructive and encouraging and ultimately drove us to the right direction.

We express our thanks to project co-ordinators **Mr.** **Sallauddin Md, Asst. Prof., and R.Ashok Asst. Prof.** for their encouragement and support.

Finally, we express our thanks to all the teaching and non-teaching staff of the department for their suggestions and timely support.

D.THARUN

G.PAVAN KUMAR

T.RITHWIK

B.VASU

**ABSTRACT**

The proposed decentralized certificate verification system that uses blockchain technology can potentially solve the issues mentioned above. By using a decentralized system, the data would not be stored on a single server that could be hacked or lost if the system crashes. Instead, the data would be stored on multiple nodes in the network, making it more secure and less vulnerable to attacks. In addition, the use of blockchain technology provides anti-tampering mechanisms, making it difficult for anyone to forge or manipulate certificates. Each certificate would be cryptographically connected to the previous certificate in the chain, creating a tamper-proof system that is difficult to hack. Moreover, the proposed system would provide faster certificate verification and issuance. Since the data would be stored on multiple nodes, verification could be done quickly without the need for human intervention. This would reduce the time and resources required for verification and issuance, making the process more efficient and cost-effective. Overall, the proposed decentralized certificate verification system that uses blockchain technology can potentially solve the issues associated with the current certificate issuance, verification, and validation process. By using a decentralized system with anti-tampering mechanisms, the system can provide a more secure and efficient way of verifying and issuing certificates, reducing the time and cost required for the process.

**TABLE OF CONTENTS**

**TITLE PAGE NO**

1. INTRODUCTION 1

* 1. PROBLEM DEFINITION
  2. OBJECTIVE OF PROJECT

1. BLOCKCHAIN FEATURES 3
2. REQUIREMENTS 4
   1. HARDWARE REQUIREMENTS
   2. SOFTWARE REQUIREMENTS
3. LITERATURE SURVEY 5
4. BLOCKCHAIN TECHNOLOGY 8
   1. ETHEREUM
   2. HYPERLEDGER
   3. SOLIDITY
   4. PYTHON
   5. SMART CONTRACTS
5. DESIGN REUIREMENTS AND UML DIAGRAMS 11
6. ALGORITHM 19
7. SOFTWARE TESTINGS 20
8. IMPLEMENTATION 23
9. RESULTS 27
10. CONCLUSION 29
11. FUTURE SCOPE 30

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **s.no** | **Figure name** | **Page no** |
| 1 | Blockchain Technology | 8 |
| 2 | UML data flow diagrams- school | 12 |
| 3 | Data flow diagram-student | 13 |
| 4 | Use case diagram | 14 |
| 5 | Static structure of a software system | 15 |
| 6 | State of a software system | 15 |
| 7 | State diagram | 16 |
| 8 | Flow of activities and actions in a system | 17 |
| 9 | objects in the system represented as boxes | 18 |
| 10 | Interactions between objects in a system | 19 |
| 11 | Blockchain hash algorithm | 19 |

# 1.INTRODUCTION

## 1.1 PROBLEM DEFINITION

## These essential credentials play a vital role in an individual’s life, and a secure and efficient process for issuing and sharing them is essential. The use of blockchain technology can provide a secure and tamper proof solution for these critical documents, ensuring that they remain trustworthy and valuable throughout an individual’s lifetime.

**1.2 OBJECTIVE OF PROJECT**

To enhance security of the issued credentials, educational institutes make use of numerous methods like assigning unique identification number, putting uniquely distinguishable hologram, affixing student’s passport sized photograph, printing the details of the students like date of birth, place of birth, parents’ name, registration/enrollment number, etc. Moreover, at the time of recruitment process, companies also need to verify the credentials that it receives directly from the applicants. Indeed, many times, companies contact the parent institution to endorse the credentials it has received from applicants. Such kind of process is tedious, costly and time-consuming. Some of the recent papers that have presented the benefits and the challenges of using blockchain technology in education. However, there is still a need to design a working prototype of student-credential sharing platform which can offer services for all the stakeholders in the education ecosystem.

Now we know what the problem is and one solution for this problem is using blockchain. Blockchain, in simple words it is a list of records, called blocks which are linked with each other using cryptography. Cryptography, it is basically a collection of techniques using which a secure communication can be made between two parties. I believe by now everyone is aware of Bitcoin (BTC) which is a cryptocurrency (digital currency) which came to all-time rise when a single. BTC would have a value of $60,000+(April 2021). It can be used to buy products and services but holds all the sales record in an online ledger with a strong cryptography to secure all the transactions.

Just like Bitcoin, there is another cryptocurrency called Ethereum (ETH) which was made by a programmer named Vitalik Buterin in 2015. It is the secondlargest cryptocurrency after Bitcoin. Difference between Ethereum and Bitcoin is that Ethereum is not just a crypto currency, it is a ledger technology which companies use to build new programs. Although both operate on blockchain technology, however Ethereum is far more robust and greater for innovation. These programs are made using “Smart Contracts” which is a self-executing contract made with the terms set by a buyer and seller and wrote it in a code. That code and the agreements are all distributed in a decentralized blockchain network. It means that blockchain does not store any information in a central location. Instead, it is copied and has been spread across a whole network of computers and whenever a new block adds in the blockchain, then each computer which holds information updates its blockchain to show that change. Smart contracts are developed using Ethereum’s programming language called solidity. It is an object-oriented language for implementing smart contracts.

Blockchain technology helps us in building a decentralized application that keeps all the data secure and tamper-free. In this application, the data is stored in text format to ease the implementation and testing, but once the transaction is done, the data is converted into hash values and stored in the block within the entire network. This provides security since a single bit of modification in a block should tamper all the data in the entire chain which is not possible because multiple copies are distributed in the peer network. So the integrity of the data is maintained. The proposed method is implemented and tested using ethereum test net. Whenever some data is about to be stored in the block of an ethereum blockchain, some gas value is reduced from the admin account and distributed in the network and it acts as the reward for the miners whose system acts as the data carrier of the block.

This gas is filled by spending some ether from their accounts. The Ethereum network limits gas availability in order to control infinite loops in the coding.

Using this motivation, an idea of making such application for sharing student credentials which is decentralized and secure, fast, and validate each credential in a form of transaction in Ethereum blockchain is proposed. All the stake holders will be able to communicate with each other securely in a manner of transaction in the blockchain network. For this DApp (Decentralized Application) we have stakeholders in the form of Schools, Students who will be registered in a school.

Using this motivation, an idea of making such application for sharing student credentials which is decentralized and secure, fast, and validate each credential in a form of transaction in Ethereum blockchain is proposed. All the stake holders will be able to communicate with each other securely in a manner of transaction in the blockchain network. For this DApp (Decentralized Application) we have stakeholders in the form of Schools, Students who will be registered in a school they are enrolled in, faculty members who will also be registered with a school they are employed in, Recruiters who can see the list of schools in the network and the students in each of the school. However, we don’t want just any recruiter to be able to access a student’s information, so we can add in the agreement a access request requirement which a recruiter sends to a student asking if they can access their data and once allowed then only its accessible to that particular recruiter

**2. Blockchain Features**

Blockchain technology has several important features that make it useful for a wide range of applications beyond just cryptocurrencies. Here's what they mean:

Immutability: Once data is recorded on the blockchain, it cannot be changed or deleted. This means that the information stored on the blockchain is permanent and cannot be altered, providing a high level of security and trustworthiness.

## Decentralized: The blockchain network is not controlled by any single entity, such as a government or company. Instead, it is run by a decentralized network of users, making it more secure and less vulnerable to attacks.

## Improved Security: The decentralized nature of the blockchain makes it more secure than traditional centralized systems because there is no single point of failure.

## Additionally, encryption technology is used to further enhance security.

## Distributed Ledgers: The blockchain maintains a public ledger of transactions that is distributed across the network of users. This provides transparency and accountability because all users can see the details of every transaction.

## Consensus: The blockchain network uses a consensus mechanism to ensure that all transactions are valid and accurate. This mechanism involves a group of users on the network agreeing on the accuracy of transactions, providing a high level of trust and security.

## 5) Faster Settlement: Transactions on the blockchain can be settled more quickly than traditional banking systems, which can take several days. This is because blockchain transactions are processed by a decentralized network of users, rather than a single central authority.

## 3. Requirements Engineering:

**3.1 Hardware Requirements:**

∙ Processor – i5 and above (64-bit OS).

∙ Memory – 8GB RAM (Higher specs are recommended for high performance)

∙ Input devices – Keyboard, Mouse

**3.2 Software Requirements:**

* Windows/Mac
* Vscode
* Node.js
* Python3
* Required python libraries

# 4.LITERATURE SURVEY

Research has been in progress to identify the fake documents and certificates, both paper and digital form. In this digital era, there is no proper method to curtail fake degrees by securing the marks memos on a tamperproof platform and verifying them digitally using unique ID. In this paper, a method is proposed to store, secure, and verify the credentials of graduates through blockchain technology.

There have been a few studies on using a blockchain based architecture in education system which dictates the benefits and challenges of using a blockchain based credential issuing/sharing system which could be easily accessible, verifiable, faster, and cost effective.

1. INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH VOLUME 9, ISSUE 03, MARCH 2020 ISSN 2277-8616 82

IJSTR©2020 www.ijstr.org Educational Certificate Verification System Using Blockchain Dinesh Kumar K, Senthil P, Manoj Kumar D.S. When people apply for jobs, the employer needs to check that the certificates and documents they provide are genuine. This can take a lot of time because it involves contacting universities and other institutions to verify the information. To make this process faster and more efficient, Blockchain technology can be used. By creating a shared database that stores information about certificates and documents, employers can easily access and verify the information they need. The use of cryptography ensures that the information is secure and cannot be tampered with. This means that employers can trust the information they find on the database. Overall, using Blockchain technology makes it easier and faster for employers to verify certificates and documents, which speeds up the hiring process.

[2] Efficient Certificate Management in Blockchain based Internet of Vehicles Ei Mon Cho 1, Maharage Nisansala Sevwandi Perera22020 20th IEEE/ACM

International Symposium on Cluster, Cloud and Internet Computing (CCGRID) As the Online of Vehicle (IoV) research trend continues, the privacy and security of each internet automobile has become a hot topic. This study aims to reduce the cost and improve the security of certifying documents, such as graduation certificates. The focus of this study is on a specific area called the Vehicle Public Key Infrastructure (VPKI). The VPKI is responsible for issuing and managing certificates for vehicles, such as cars, trucks, and other modes of transportation. The proposed solution uses Blockchain technology to make the process of issuing and managing certificates more secure and cost-effective. The solution uses activation codes that are tied to the time a vehicle is considered "nonrevoked" to validate the certificate. This means that if a vehicle is no longer in use, its certificate can be removed from the system to save costs.

[3] Design and Implementation of Work Training Certificate Verification Based On Public Blockchain Platform 1st Irawan Afrianto Informatics Engineering Department, 2nd Yayan Heryanto Informatics Engineering Department

The purpose of this research is to develop a secure and efficient system using blockchain technology for storing job training certificates. By using public blockchain, the certificates are protected from being forged or altered. Smart contracts are utilized to create data blocks that are added to the Ethereum blockchain network. The certificate files are stored in a distributed environment called Inter Planetary File System (IPFS), which provides secure and quick access. The research showed that certificate data can be safely stored on Ethereum public blockchain and supporting files on IPFS.

[4] Efficient Distributed Admission and Revocation using Blockchain for Cooperative ITS Noureddine Lasla∗, Mohamed Younis§, Wassim Znaidi∗ and

Dhafer Ben Arbia∗ ∗Qatar Mobility Innovations Center (QMIC), QSTP, Doha, Qatar.

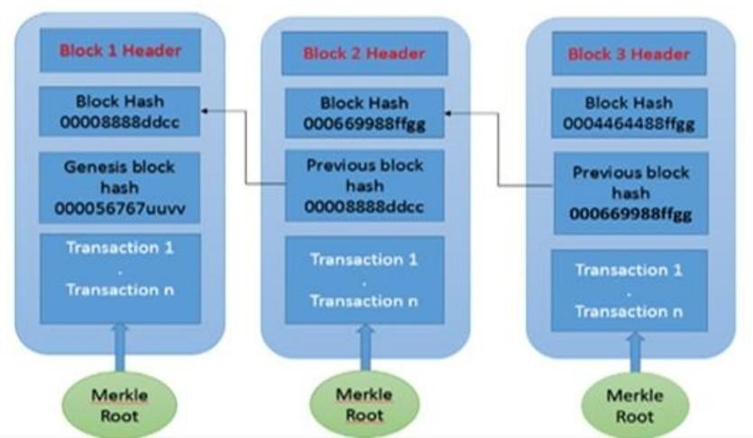
The Cooperative Intelligent Transportation System (CITS) is an innovative technology that enables vehicles to communicate with each other, enhancing road safety. However, ensuring secure communication remains a significant challenge in the research community. The existing solutions for inter-vehicle communication security mostly rely on digital certificates' authentication.

To address these challenges, this study proposes using blockchain technology to maintain distributed and immutable records of each vehicle's certificate, whether it is valid or revoked. This approach replaces certificate verification with a lightweight blockchain-based authentication solution. Additionally, the study proposes a fully distributed method for vehicle admission and revocation. The findings demonstrate that this technique can reduce computational overhead and improve performance, making it a promising solution for inter-vehicle communication security.

[5] Cerberus: A Blockchain-Based Accreditation and Degree Verification System Aamna Tariq∗ , Hina Binte Haq, Syed Taha Ali‡ School of Electrical Engineering and Computer Sciences (SEECS), National University of Sciences and Technology (NUST), Islamabad, Pakistan.

Fake certificates are a big problem and the usual way of checking them takes a lot of time and is expensive. This study proposes a solution that uses blockchain technology to verify identity, which is faster and more efficient. It also has a system to revoke certificates if they are fake. The system is easy to use and does not require special skills. A prototype has been created and it has features like data privacy, verifying transcripts, and sharing data only with those who need to see it.

1. **Blockchain Technology:**

****

## Blockchain is a distributed ledger technology that allows for the creation of decentralized and secure systems for sharing data. Each block in a blockchain contains a set of transactions, which can be anything from cryptocurrency transactions to digital certificates or bill of lading. These transactions need to be communicated to all the nodes in the blockchain network so that they can be validated and added to the ledger.

Miners validate transactions and group them together in a new block, which is then added to the distributed ledger. Each block is cryptographically sealed, making it tamper-proof and ensuring the security of the system. The database used in blockchain is distributed and records the various transactions that occur within the system.

Each block carries the hash value of its previous counterpart, which creates a link between blocks and forms a blockchain. The hash value ensures that any changes made to the previous block will be reflected in the hash value of the current block. This makes it nearly impossible to tamper with any previous transactions, as doing so would require changing the hash values of every subsequent block.

The fundamental principle of blockchain is to have a distributed ledger that is used to keep track of individual transactions. These transactions are validated by different nodes within the network and are finalized using consensus algorithms, such as permissionless or permissioned consensus algorithms. In a permissionless system, anyone can join the network and participate in the validation process. In a permissioned system, only authorized participants can validate transactions.

**5.1 Ethereum**

Ethereum is a technology that uses a special type of database called a blockchain to store and manage information. It allows people to create and run applications that work without a third party. This technology also has its own digital currency called ether, and it can be used to create smart contracts, which are computer programs that automatically execute the terms of a contract.

**5.2 Hyperledger**

Hyperledger is an open-source project hosted by the Linux Foundation that provides a framework and tools for building blockchain-based applications and solutions, particularly for enterprises. Unlike public blockchains like Bitcoin or Ethereum, Hyperledger is designed to support permissioned blockchains—networks where participants are known, authenticated, and authorized.

**5.3 Solidity**

Solidity is a programming language used to create smart contracts on the Ethereum blockchain. Smart contracts are like digital agreements that automatically enforce rules and handle transactions. Solidity is similar to other programming languages like C++, Python, and JavaScript, and allows developers to create complex contracts with advanced features. With Solidity, you can create contracts for various purposes like voting, crowdfunding, auctions, and digital wallets. It's important to use the latest version of Solidity when creating contracts because it has new features and bug fixes.

**5.4 Python**

Python could be a backend programming dialect that’s extraordinary for apprentices. Python is comparative in many ways to Ruby but is less wordy than other programming dialects - a small less wordy. Python is congenial. Indeed, if you haven’t taken a CS lesson, you'll still compose a valuable apparatus in Python. It’s high-level, so you don’t have to bargain with the lower-level angles of programming such as memory management. Python can be utilized for scripting, web scratching, and making information sets. It’s popular in the scientific community for logical computing; there are libraries that make it easy to share academic code ventures in Python. Python could be a web programming dialect, so it interfaces with the web. It knows how to get and send web demands and conversation to databases. Python is said to be “loosely typed”. This category of programming dialects doesn’t require you to state the type of esteem a work returns after you characterize the work or the sort of variable you recently created.

* 1. **Smart Contracts**

**Private Smart Contract**

Permissioned blockchains have become increasingly popular in the industry as they offer an efficient way to conduct business transactions within a closed network of stakeholders. This is achieved by using a private blockchain with a limited number of participants, unlike a public blockchain, where the cost of validation can be significantly high due to the energy spent on the consensus process for proof of work. Private blockchains can be efficient when the number of nodes is small, and the transactions are faster.

The consensus process used in public and private blockchains differ significantly.

Proof of work and proof of stake are the consensus algorithms used in public blockchain, while private blockchain uses other consensus algorithms. IBM has developed a range of blockchain development environments for business services, such as Hyperledger fabrics, Hyperledger composer, Hyperledger Indy, and Hyperledger Sided. The members of the Hyperledger Fabric are permitted blockchain networks that participate in the development of the Hyperledger Fabric. The member organizations in the network are responsible for assigning peers to network participants. Additionally, each peer in the network is certified by a certificate authority.

**Public smart contract**

In a permissionless blockchain, anyone can participate and deploy smart contracts without any restrictions. However, to prevent abuse, users need to pay a small fee to create and run smart contracts. In blockchain applications like Bitcoin, these smart contracts are created using something called "bitcoin scripts", which define the terms and conditions of the contract.

1. **Design Requirements :**

**Concept of UML :**

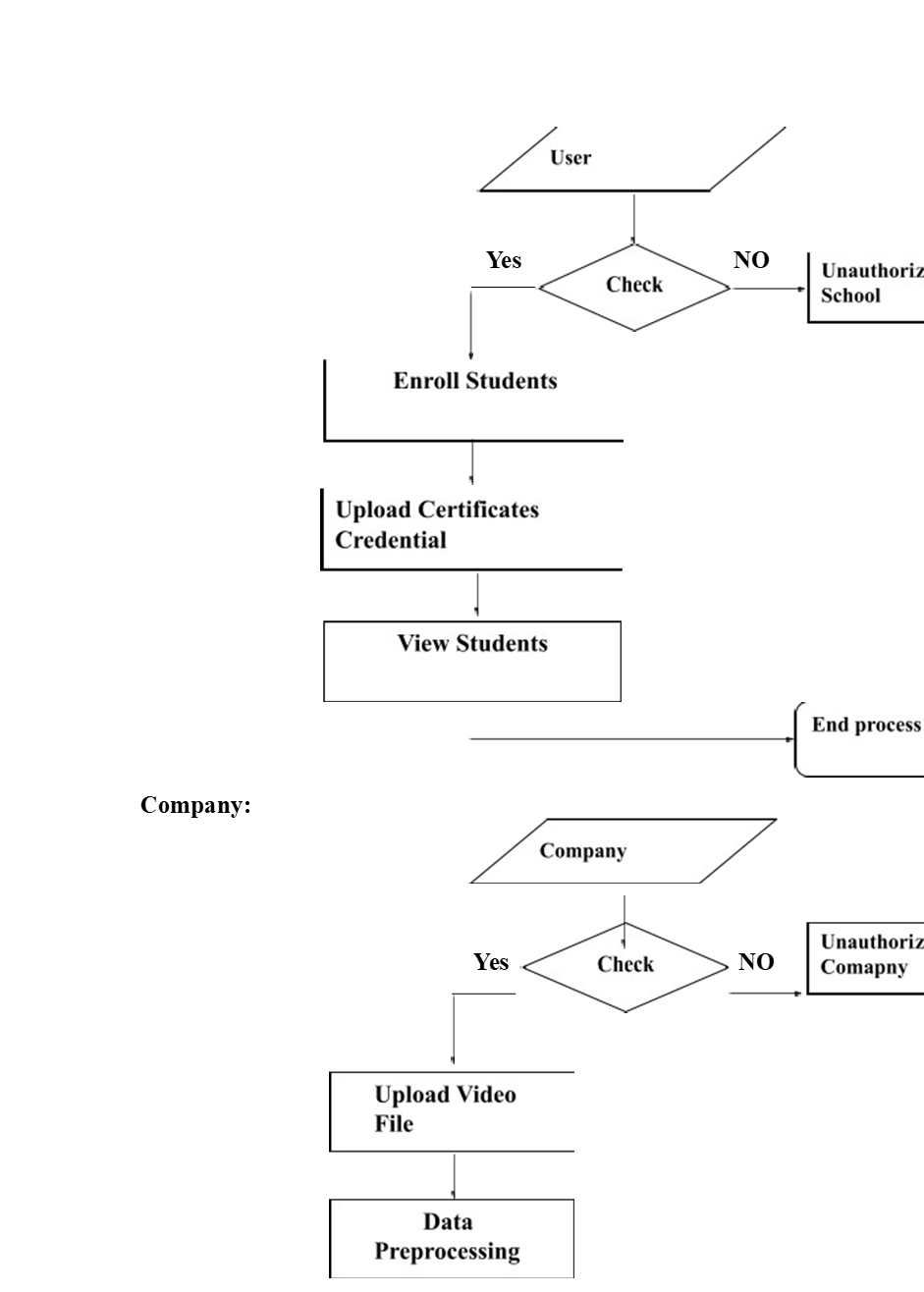
The Unified Modelling Language (UML) is the standard demonstrating dialect for computer program and frameworks improvement. In the framework plan, you show for one vital reason: to oversee complexity. Displaying makes a difference when you see the timberland for the trees, permitting you to centre on, capture, report, and communicate the critical perspectives of your system's plan.

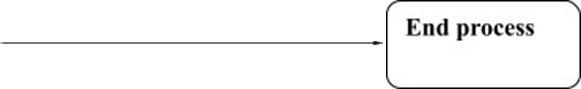
**UML DIAGRAMS**:

**DATA FLOW DIAGRAM:**

1. School: First the School administrator registers on the website by signing up. So, when he logs in, the website checks whether the details are correct and allows him to access it. Then the school administrator Enrolls the students. Then he uploads the Certificate credentials. He can also view these certificates. The process is ended.
2. Company: First the Company administrator registers on the website by signing up. He does this with the help of the school administrator. When he logs in he can send the request access to the student and also view all these access requests.
3. Student: The student logs in with the help of id that is given by the school administrator while enrolling the student. He logs in and can check the certificates he received. He can also check the access requests from companies. He can approve or disapprove them, So, these are all the activities that a student can do in the website.

**School:**

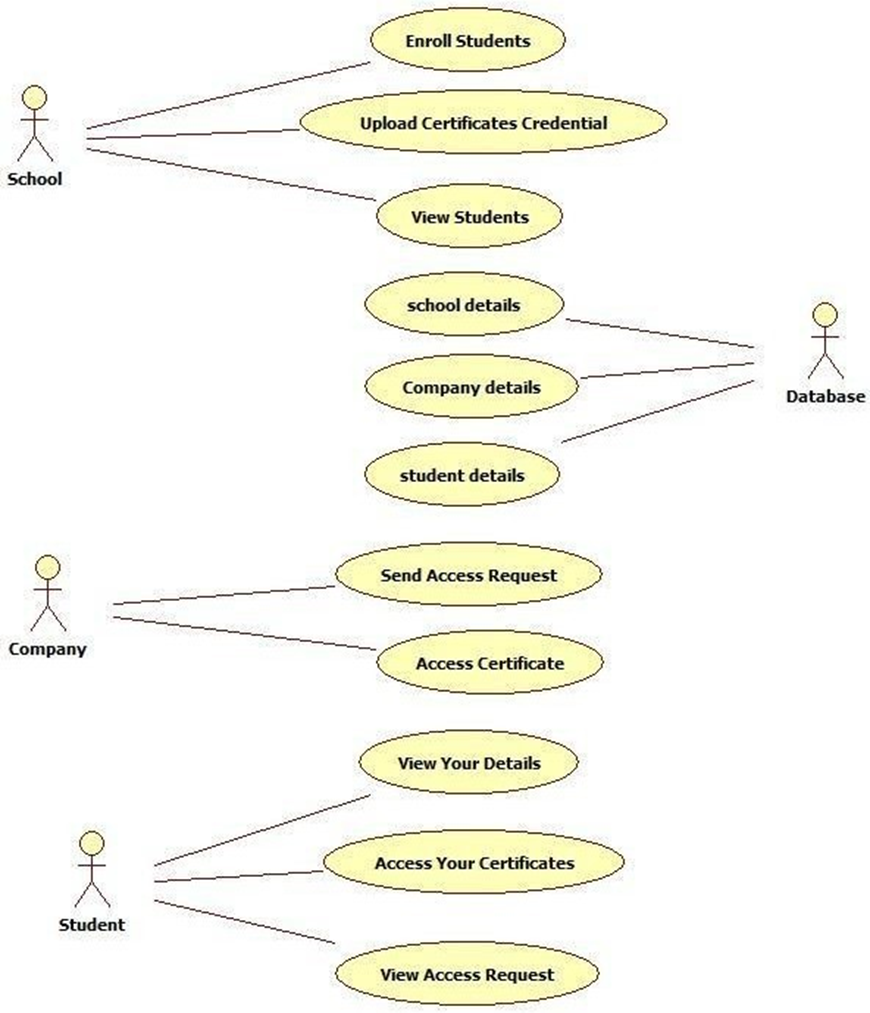
****

****

## Student:

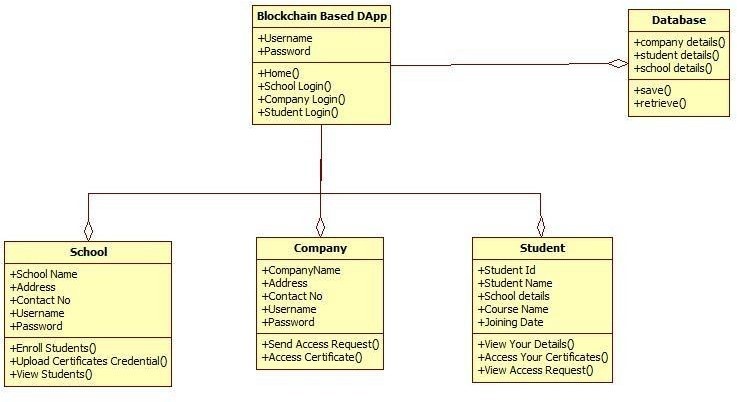
## 

**Use case Diagram:**



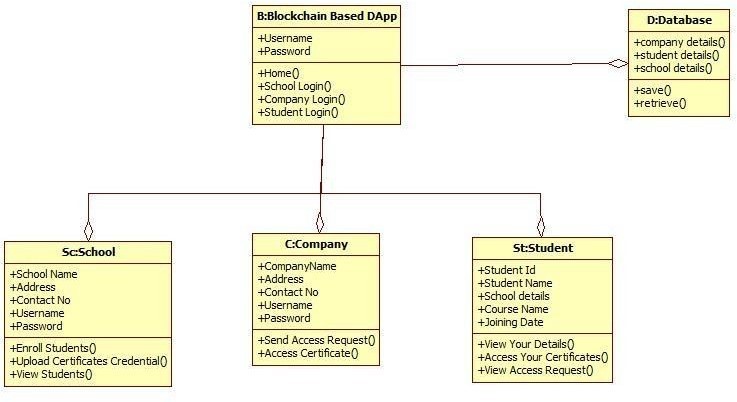
There are four actors in this use case diagram. They are school, Company, Student and Database. The school will enroll students, Upload certificate credentials, View students. The Company will send access requests, access certificates. The student will access his/her certificates, View Access requests.

**Class diagram:**



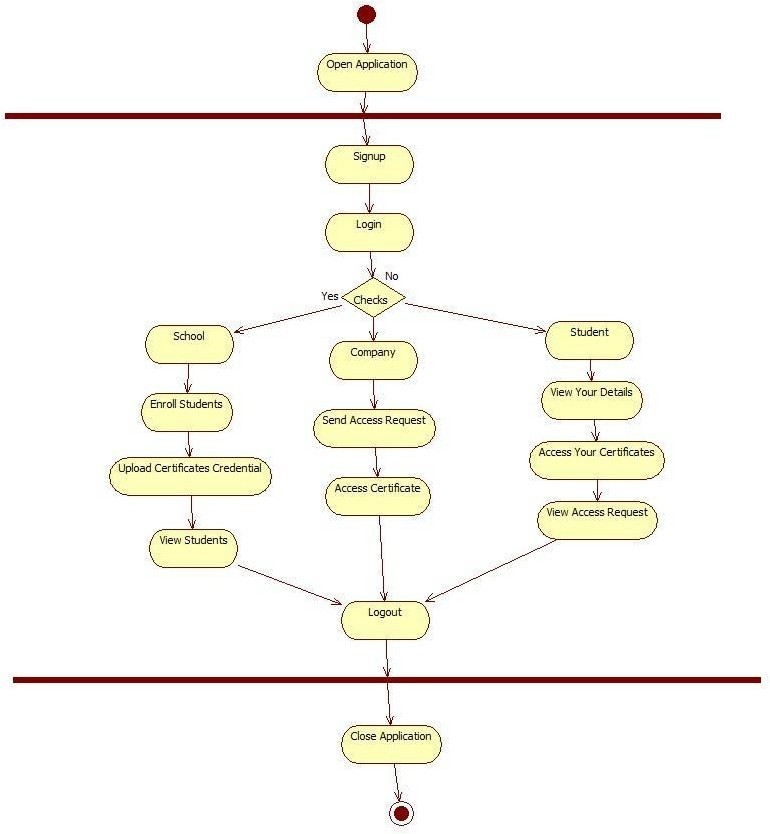
The class diagram is a fundamental tool used in software engineering to represent the static structure of a software system. It captures the key concepts and relationships between them in a graphical format, providing a clear and concise representation of the system's structure. The relationships or associations between classes can be either "is-a" or "has-a" relationships. An "is-a" relationship represents a generalization/specialization hierarchy between classes, where a subclass inherits properties and methods from its parent class. A "has-a" relationship represents a composition or aggregation relationship between classes, where one class contains another as a component.

**Object diagram:**



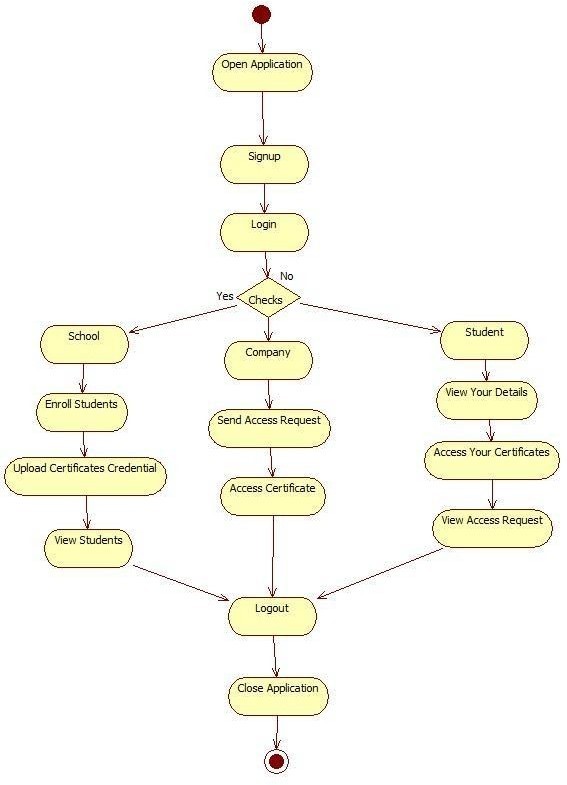
An object diagram is a type of diagram that shows the state of a software system at a particular moment in time. It represents the objects in the system as they exist and interact with each other at that moment. In software development, a class is like a blueprint for creating objects. An object, on the other hand, is an instance or a copy of that blueprint. So, if a class is a recipe for making a cake, an object would be an actual cake made using that recipe.

**State diagram:**



First the user opens the application. Then he signs up on the website. After signing up he logs on to the website. If the user is a School. He enrolls the students, uploads the certificate credentials, and views the students. If the user is a company, he sends access requests and access certificates of various students. If the user is a student. He accesses his/her certificates and views access requests sent by various companies.

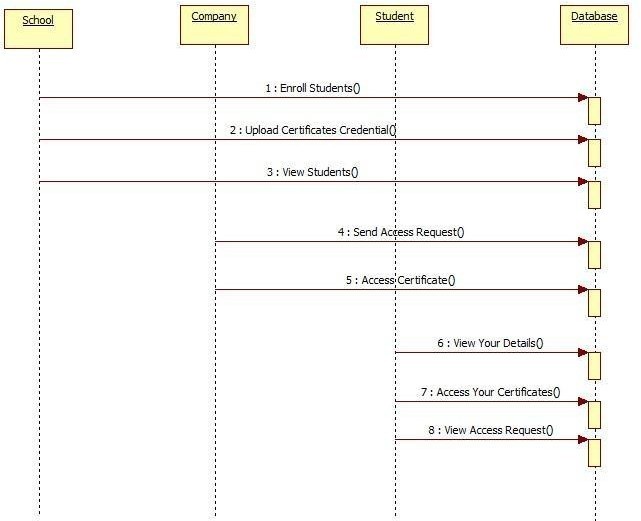
**Activity diagram:**



The activity diagram is a type of diagram used in software engineering to describe the flow of activities and actions in a system. It captures the sequence of activities and decisions that occur in the system to achieve a specific goal.

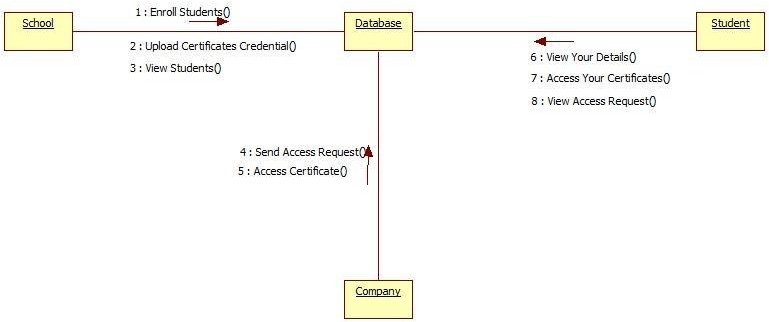
An activity is a task or action that takes place in the system, and it can be represented by a rounded rectangle in the diagram. An action is a specific task that is performed during an activity, and it can be represented by a rectangle in the diagram.

**Sequence diagram:**



In a sequence diagram, the different objects in the system are represented as boxes or lifelines, which run vertically down the diagram. The interactions between the objects are represented by horizontal arrows, which are also called "messages". These arrows show the flow of communication between the objects, indicating which object is sending the message and which object is receiving it.

**Collaboration diagram:**

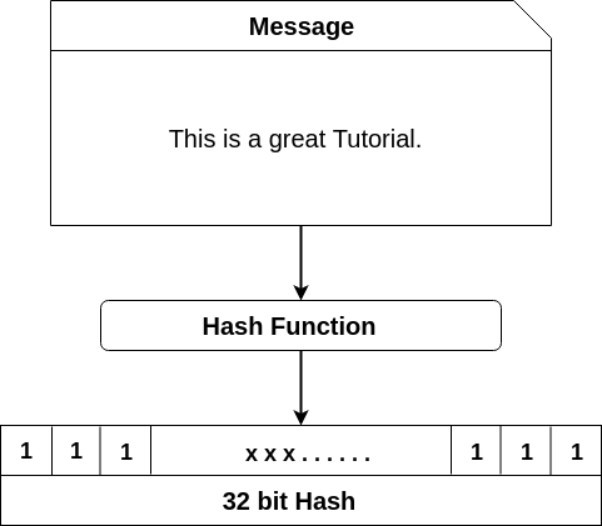


## A collaboration diagram is a type of diagram used in software engineering to show the interactions between objects in a system. The interactions between objects are represented by numbered interactions, which help to trace the sequence of the interactions. Each interaction is labeled with a number, which represents the order in which the interactions occur. The numbers make it easier to understand the flow of interactions between the objects.

1. **Algorithm:**

## Blockchain Hash Function

## A hash function is a mathematical algorithm that takes an input string of any length and transforms it into a fixed-length output, which is called a hash. The fixed length output can vary depending on the hash function used. Hash functions are widely used in cryptography or digital signatures, data integrity checks, and authentication.



In contrast to encryption algorithms, where you can decrypt the message back to its original form using a key, with a hash function, once you generate a hash, you cannot reverse it back to the original message. This makes it an irreversible process.

In the context of cryptocurrencies like Bitcoin, the blockchain uses cryptographic hash functions in its consensus mechanism. The transactions are taken as an input and run through a hashing algorithm, which produces a fixed-length output. This output serves as a digital fingerprint or hash of the transaction data, and it is used to uniquely identify each transaction in the blockchain. The properties of hash functions, such as uniqueness and one-way computation, make them ideal for ensuring the security and integrity of the blockchain.

1. **Software Testing:**

## Testing is a way of finding mistakes or problems in something, like software or a product. It helps make sure that the thing works as expected and doesn't have any major problems that could cause issues for users. There are different types of tests that focus on different aspects of the thing being tested. By doing these tests, we can make sure that the thing we're testing meets the requirements and works correctly.

**TYPES OF TESTS**

**Unit Testing**

Unit testing is a way of testing individual parts of a software program to make sure that they work correctly. It involves testing each component of the program separately to make sure that it produces the right output for a given input. This type of testing is done after a single part of the program has been completed, but before all the parts are put together. Unit testing is important because it helps to catch mistakes early on, before they can affect other parts of the program. By testing each part separately, we can make sure that the program works as it's supposed to, and that all the different parts work together correctly.

**Integration testing**

Integration testing is a way of testing how different parts of a software program work together. It involves testing the program to make sure that all the different parts work together correctly. This type of testing is done after each individual part of the program has been tested separately. Integration testing is important because it helps to catch mistakes that can happen when different parts of the program are combined. By testing how the program works, we can make sure that it runs smoothly and that all the different parts work together correctly. Integration tests help to ensure that the program is reliable and works as expected.

**Functional test**

Functional tests are designed to check if the software functions as expected according to the requirements, system documentation, and user manuals. These tests check if the software can handle valid inputs and reject invalid inputs, perform the required functions, produce the expected outputs, and work well with interfacing systems or procedures. To prepare for functional tests, testers need to identify the business process flows, data fields, and predefined processes to be tested. They also need to check if the current tests are effective and if additional tests are needed. Overall, functional testing helps ensure that the software meets the expected business and technical requirements.

**System Test**

System testing is the process of testing the entire software system to make sure it meets all the requirements and works as expected. This type of testing is used to ensure that the final product is fully functional and meets the needs of its users. It focuses on how different parts of the system work together, and tests are designed to make sure that all components are integrated properly, and that the software behaves as expected in various scenarios. An example of system testing could be testing the software configuration to ensure that it works as expected, and that all parts of the system are linked and integrated correctly.

**White Box Testing**

White box testing is a testing technique in which the tester has knowledge of the internal workings, structure, and code of the software being tested. This knowledge allows the tester to design test cases that target specific areas of the code and evaluate the functionality of those areas. White box testing is particularly useful for identifying defects in the code logic and verifying that the code meets the expected standards. In this type of testing, the tester has access to the source code and can review the code to understand how it works. The tester can also use code analysis tools to identify potential problems and to verify that the code adheres to coding standards.

**Black Box Testing**

Black box testing is a way of testing software where the tester doesn't know anything about how the software was made or how it works inside. The tester uses a list of instructions or requirements to test the software, like a recipe to make a cake. The tester checks if the software does what it is supposed to do, without worrying about how it was made or what is inside it.

**Unit Testing**

Unit testing is a way of testing individual parts, or "units," of a software program to make sure they work correctly. This testing is usually done by the same person who wrote the code. Field testing is testing the software in real-world situations. It's done by people who are not involved in writing the code. The objectives of testing are to make sure that all the data that's entered into the program is handled correctly, all the links in the program work as they should, and there are no delays or errors in the program's responses. The features that will be tested are whether data is entered correctly, whether there are any duplicates, and whether all the links take the user to the right place.

The test strategy for this particular software includes manual field testing and detailed functional testing. The objectives of the tests are to ensure that all field entries are functioning properly, pages are activated correctly from the identified links, and that the entry screen, messages, and responses are not delayed.

To achieve these objectives, the following features will be tested:

1. Verification of the format of the entries to ensure they are correct
2. Prevention of duplicate entries to maintain data integrity
3. Checking of links to ensure that they are taking the user to the correct pages.

These tests will help to identify any issues with the software early in the development process, allowing developers to address them before the software is released to the public. By isolating the individual components and testing them thoroughly, unit testing can improve the overall quality and reliability of the software application.

1. **IMPLEMENTATION**

pragma solidity >=0.7.0 <0.9.0;

contract Verification {

constructor() { owner = msg.sender; }

uint16 public count\_Exporters =0;

uint16 public count\_hashes=0;

address public owner;

struct Record {

uint blockNumber;

uint minetime;

string info;

string ipfs\_hash;

}

struct Exporter\_Record{

uint blockNumber;

string info;

}

mapping (bytes32 => Record) private docHashes;

mapping (address => Exporter\_Record) private Exporters;

//---------------------------------------------------------------------------------------------------------//

modifier onlyOwner() {

if (msg.sender != owner) {

revert("Caller is not the owner"); }\_; }

modifier validAddress(address \_addr) {

assert(\_addr != address(0)); \_; }

modifier authorised\_Exporter(bytes32 \_doc){

if (keccak256(abi.encodePacked((Exporters[msg.sender].info )))!= keccak256(abi.encodePacked((docHashes[\_doc].info))))

revert("Caller is not authorised to edit this document");

\_; }

modifier canAddHash(){

require(Exporters[msg.sender].blockNumber!=0,"Caller not authorised to add documents"); \_; }

//---------------------------------------------------------------------------------------------------------//

function add\_Exporter(address \_add,string calldata \_info) external

onlyOwner(){

assert(Exporters[\_add].blockNumber==0);

Exporters[\_add].blockNumber = block.number;

Exporters[\_add].info = \_info;

++count\_Exporters;

}

function delete\_Exporter(address \_add) external

onlyOwner

{

assert(Exporters[\_add].blockNumber!=0);

Exporters[\_add].blockNumber=0;

Exporters[\_add].info="";

--count\_Exporters;

}

function alter\_Exporter(address \_add,string calldata \_newInfo) public

onlyOwner()

{

assert(Exporters[\_add].blockNumber!=0);

Exporters[\_add].info=\_newInfo; }

function changeOwner(address \_newOwner) public

onlyOwner validAddress(\_newOwner) { owner = \_newOwner; }

event addHash(address indexed \_exporter,string \_ipfsHash);

function addDocHash (bytes32 hash,string calldata \_ipfs) public

canAddHash

{

assert(docHashes[hash].blockNumber==0 && docHashes[hash].minetime==0);

Record memory newRecord =

Record(block.number,block.timestamp,Exporters[msg.sender].info,\_ipfs);

docHashes[hash] = newRecord;

++count\_hashes;

emit addHash(msg.sender,\_ipfs);

}

function findDocHash (bytes32 \_hash)

external view returns (uint,uint,string memory,string memory) {

return (docHashes[\_hash].blockNumber,docHashes[\_hash].minetime,docHashes[\_hash].info,docHashes[\_hash].ipfs\_hash );

}

function deleteHash (bytes32 \_hash) public

authorised\_Exporter(\_hash)

canAddHash

{

assert(docHashes[\_hash].minetime!=0);

docHashes[\_hash].blockNumber=0;

docHashes[\_hash].minetime=0;

--count\_hashes;

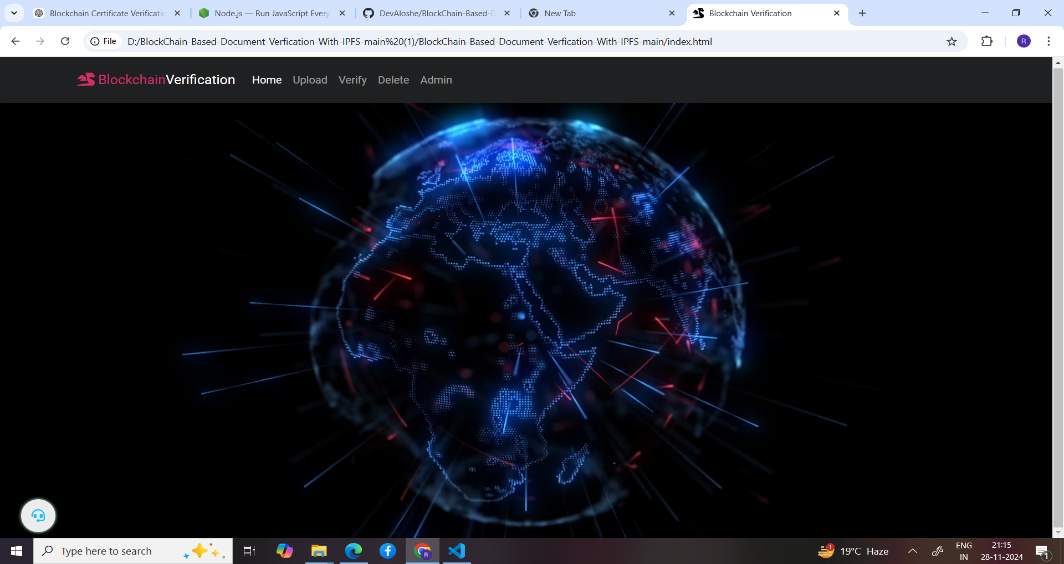
}

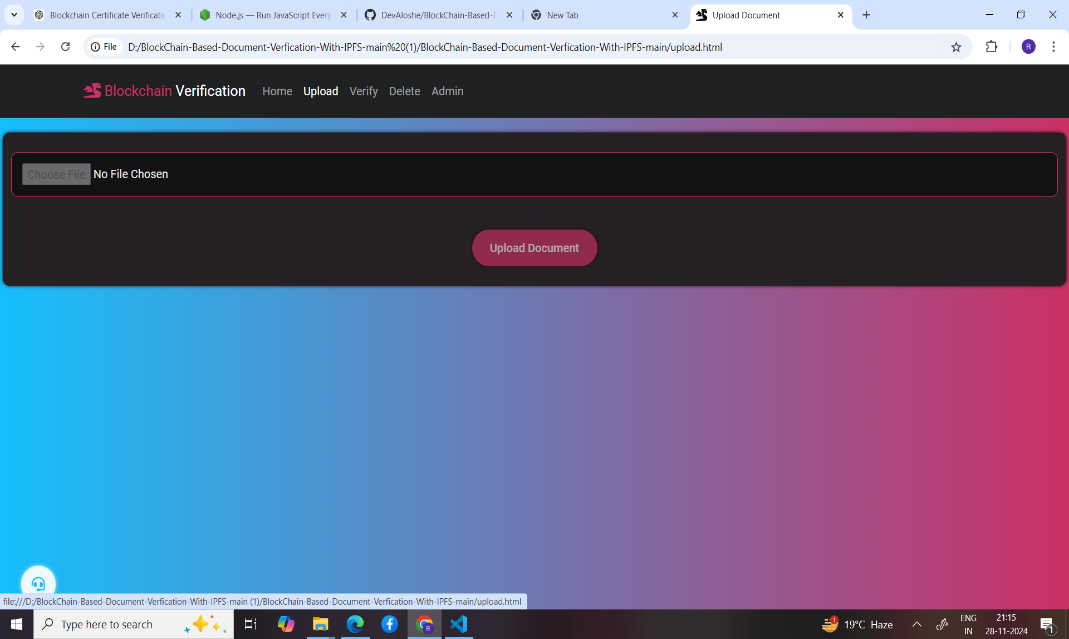
function getExporterInfo(address \_add) external view returns(string memory){

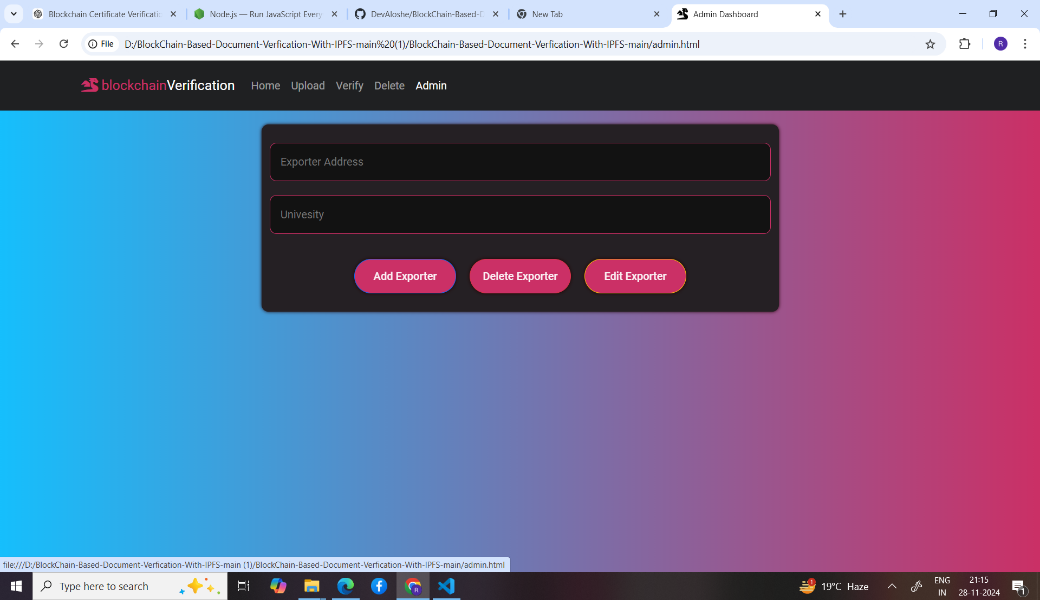
return (Exporters[\_add].info);

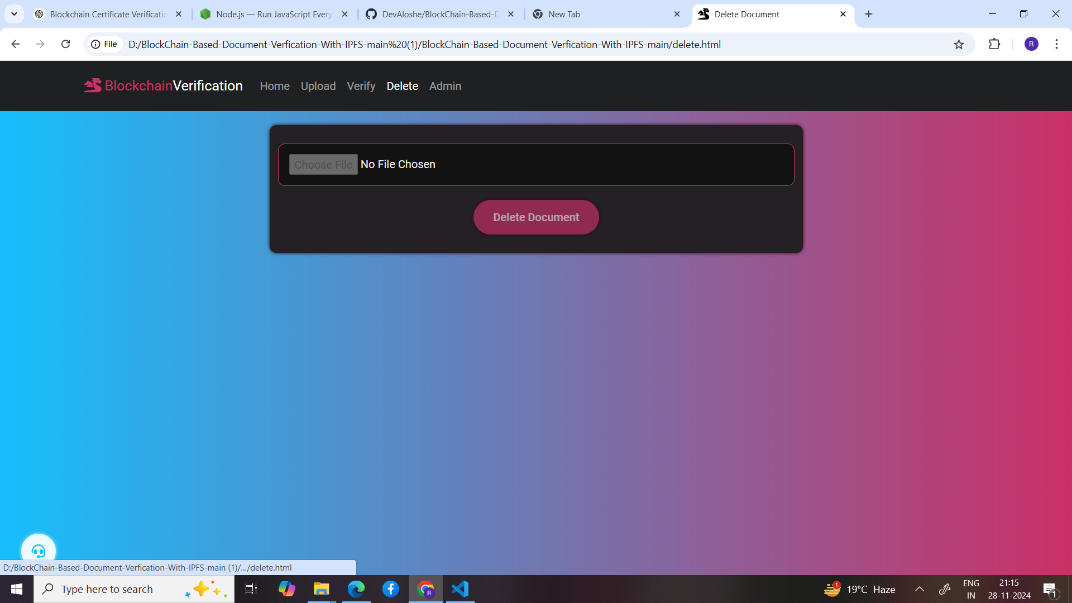
}

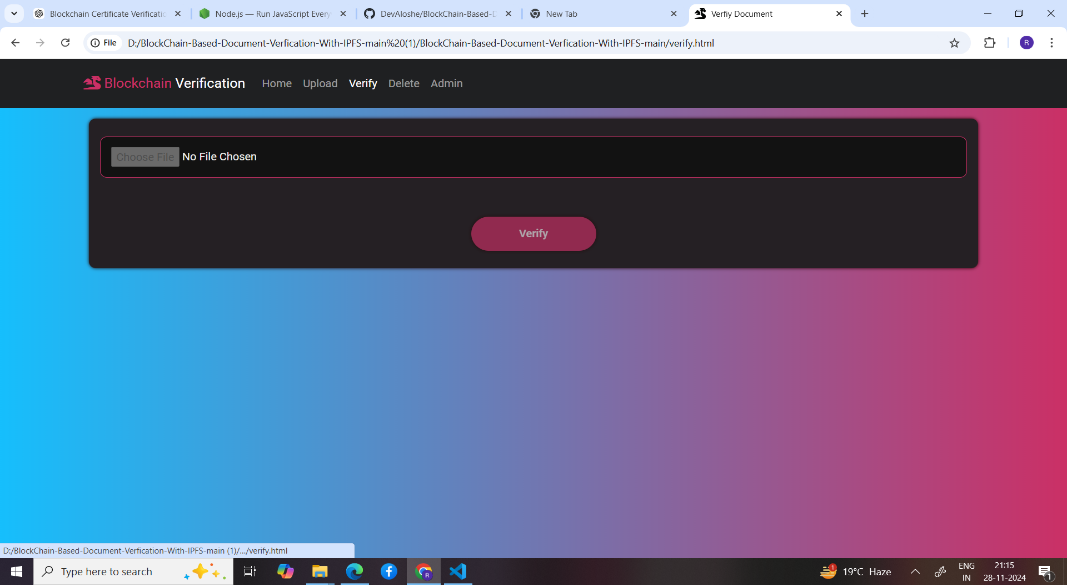
}

 **10. RESULTS**

****

****



****

**11. CONCLUSION**

• In this Research Work, a Blockchain-based Educational Certificate Verification decentralized application ramed DiUcerts was developed in the Ethereum Blockchain platform.

• The first objective of this research work was achieved by the progressive and effective implementation of the proposed model and also developed DApp.

• This paper is introduced to avoid the middleman or any third party in the certificate verification process.

• One of our goals is to diminish the cost and time of the verification process. Surprisingly, this paper evaluates the cost of the proposed decentralized app, and it cost less than a dollar without any server maintenance cost.

• However, the main challenge of this paper is to eradicate fake educational certificates and our DApp almost met all the conditions to fight against this fraud and scam channel.

• In conclusion, from this work, this paper encourages that developers should be considered the Ethereum Blockchain platform for developing decentralized applications on Blockchain and it will turn into an interesting sector for the upcoming generation.

**12 . Future Scope**

1. Blockchain solutions can enable global interoperability, allowing educational institutions worldwide to share verified records, facilitating cross-border education and employment verification.

2. Advanced security measures like quantum-resistant protocols and AI-based authentication can ensure the long-term integrity of educational credentials.

3. Integration with job platforms and government systems can streamline recruitment processes and enable seamless authentication of certificates for public services.

4. The system can expand to include certifications from online courses, skill assessments, and informal education, making it versatile and inclusive.

5. Smart contracts can automate processes such as scholarship disbursements and certificate issuance, reducing administrative overhead.

6. Real-time verification improves efficiency while reducing delays, and self-sovereign identity systems empower students to manage and share their credentials securely.

7. Green blockchain initiatives and legislative support for legal recognition of blockchain-certified credentials can make the system more sustainable and widely accepted.