

**A PROJECT REPORT ON**

**VERIFICATION OF CERTIFICATES USING BLOCKCHAIN**

*SUBMITTED TO*

**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY,  
KAKINADA**

*Submitted in partial fulfillment of the requirements for the award of the degree*

**BACHELOR OF TECHNOLOGY**

**In**

**ELECTRONICS AND COMMUNICATION ENGINEERING**

*By*

<b>R.RVV SAI</b>	<b>- 198X1A04C7</b>
<b>M.LSV SUBRAHMANYAM</b>	<b>- 198X1A0499</b>
<b>J. PAVAN KUMAR</b>	<b>- 198X1A0468</b>
<b>K. BALAJI</b>	<b>- 198X1A0476</b>
<b>K. VAMSI</b>	<b>- 198X1A0471</b>

**UNDER THE GUIDENCECE OF**

**MR.K. SRINIVAS RAO M.Tech,(PhD)**



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING**

**KALLAM HARANADHAREDDY INSTITUTE OF TECHNOLOGY**

**AN ISO 9001:2015 CERTIFIED INSTITUTION**

**ACCREDITED BY NBA & NAAC WITH 'A' GRADE**

**(APPROVED BY AICTE, AFFILIATED TO JNTUK, KAKINADA)**

**NH-16, CHOWDAVARAM, GUNTUR – 522019**

**(2019-2023)**

# **KALLAM HARANADHAREDDY INSTITUTE OF TECHNOLOGY**

Approved by AICTE, New Delhi; Affiliated to JNTUK, KAKINADA

Accredited by NAAC with 'A' Grade & NBA

NH-16, Chowdavaram, Guntur-522019

## **DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING**



## **CERTIFICATE**

This is to certify that the major project report entitled **“VERIFICATION OF CERTIFICATES USING BLOCKCHAIN”** is submitted by **R.RVV SAI (198X1A04C7), M.LSV SUBRAHMANYAM (198X1A0499), J. PAVAN KUMAR (198X1A0468), K. BALAJI (198XX1A0476), K. VAMSI (198X1A0471)** to the Jawaharlal Nehru technological university Kakinada. Kakinada in partial fulfilment for the award of degree of Bachelor of Technology in Electronics and communication Engineering is Bonafede record of the project work carried out by them under my supervision during the academic year 2022-2023.

### **PROJECT GUIDE**

**MR.K.SRINIVAS RAO**  
M.Tech, (PhD)  
Associate Professor, ECE, KHIT

### **HEAD OF THE DEPARTMENT**

**Dr.S. SURYANARAYANA**  
B. TECH, M. TECH, PhD  
Professor & HOD

**External Examiner**

## DECLARATION

We here declare that this project entitled “**VERIFICATION OF CERTIFICATES USING BLOCKCHAIN**” been independently carried out by us. As per my knowledge, no part of this work has submitted for any degree in any institution, university and organization previously.

- |                       |              |
|-----------------------|--------------|
| 1. R.RVV SAI          | - 198X1A04C7 |
| 2. M.LSV SUBRAHMANYAM | - 198X1A0499 |
| 3. J. PAVAN KUMAR     | - 198X1A0468 |
| 4. K. BALAJI          | - 198X1A0476 |
| 5. K. VAMSI           | - 198X1A0471 |

## ACKNOWLEDGEMENTS

We proudly express our gratitude and respect towards our honorable chairman **SRI KALLAM MOHAN REDDY**, Chairman of Kallam group for his precious support in the college.

We are thankful to **Dr. M. UMASANKAR REDDY**, Director for his encouragement and support for the completion of project we are inspired a lot through his valuable message.

We express our great pleasure to **Dr. B. S. B. REDDY**, Principal, KHIT, Guntur, for his support during and till the completion of the project.

We are really thankful to **Dr. S. SURYANARAYANA**, Head of the Department of **ECE**, for providing the laboratory facilities to the fullest extent as and when required and also for giving us the opportunity to carry out the project work in the college.

We are really thankful to our project guide "**MR.K. SRINIVAS RAO**", Associate Professor, KHIT, Guntur, for her excellent guidance right from selection of the project and her valuable suggestions throughout the project work

We are thankful to all teaching and non-teaching of Electronics and Communication Engineering department and management and my friends for their direct and indirect work provided to use in completing the project.

<b>R.RVV SAI</b>	<b>- 198X1A04C7</b>
<b>M.LSV SUBRAHMANYAM</b>	<b>- 198X1A0499</b>
<b>J. PAVAN KUMAR</b>	<b>- 198X1A0468</b>
<b>K. BALAJI</b>	<b>- 198X1A0476</b>
<b>K. VAMSI</b>	<b>- 198X1A0471</b>

# TABLE OF CONTENTS

Project report

Certificate

Declaration

Acknowledgment

Abstract

<b>TITLE</b>	<b>PAGE NO</b>
<b>1. INTRODUCTION</b>	
1.1 INTRODUCTION	10
1.2 OVERFLOW	10-11
1.3 PROBLEMS OR ISSUES BEING SOLVED	11
1.4 OBJECTIVE	12
1.5 DESCRIPTION	12
<b>2. LITERATURE SURVEY</b>	
2.1 LITERATURE SURVEY	13
<b>3.SYSTEM ANALYSIS &amp; FEASIBILITY</b>	
3.1 EXISTING SYSTEM	14
3.1.1 DISADVANTAGES	14
3.2 PROPOSED SYSTEM	14
3.2.1 ADVANTAGES	15
3.3 FEASIBILITY	15
3.3.1 ECONOMICALLY FEASIBILITY	15

3.3.2 TECHNICAL FEASIBILITY	15
3.3.3 SOCIAL FEASIBILITY	15

## **4. SYSTEM REQUIREMENTS**

4.1 SOFTWARE REQUIREMENTS	16
4.2 HARDWARE REQUIREMENTS	16
4.3 NON-FUNCTIONAL REQUIREMENTS	16

## **5. SYSTEM DESGIN**

5.1 UML DIAGRAM	17
5.1.1 SYSTEM ARCHITECTURE	18
5.1.2 WORKFLOW	18-19
5.1.3 USE CASE DIAGRAM	19
5.1.4 CLASS DIAGRAM	20
5.1.5 SEQUENCE DIAGRAM	20
5.1.6 COMPONENT DIAGRAM	21

## **6. IMPLEMENTATION**

6.1 MODULE DESCRIPTION	22
6.2 ALGORITHM	23
6.3 IMPLEMENTATION PROCEDURE	22-23

## **7. SYSTEM ENVIRONMENT**

7.1 PYTHON	24
7.1.1 ADVANTAGES OF PYTHON	24-25
7.1.2 ADVANTAGES OF PYTHON OVER OTHER LANGUAGE	25-26
7.1.3 DIADVANTAGES OF PYTHON	26-27
7.1.4 HISTORY OF PYTHON	27-35

7.1.5 PYTHON DEVELOPMENT STEPS	35-40
7.1.6 HOW TO INSTALL PYTHON ON WINDOWS & MAC	40-46
7.2 BLOCKCHAIN	46-47
7.2.1 BLOCKCHAIN WORKS	47-48
7.2.2 BULIDING TRUST WITH BLOCKCAHIN	48
7.2.3 BENFITS OF BLOCKCHAIN	49-50
7.3 HASH VALUE	50-51
7.4 GUI APPLICATIONS AND TYPES	51-57
7.5 PICKLE MODULE	57-58
<b>8. SYSTEM TEST</b>	
8.1 TYPES OF TESTS	59
8.1.1 UNIT TEST	59
8.1.2 INTEGRATION TEST	59
8.1.3 FUNCTIONAL TEST	60
8.1.4 SYSTEM TEST	60
8.1.5 WHITEBOX TEST	60
8.1.6 BLACKBOX TEST	61
8.1.7 UNIT TEST	61
8.18 TEST STRATEGY & APPROACH	61
8.1.9 INTEGRATION TEST	61
8.1.10 ACCEPTANCE TEST	62
<b>9. PROJECT OUTCOME</b>	
9.1 PROJECT OUTCOME	63-66
<b>10. CONCLUSION</b>	
10.1 CONCLUSION	67
10.2 FUTURESCOPE	68

## **11. REFERENCES**

### 11.1 REFERENCES

69



## LIST OF FIGURES

1. SYSTEM ARCHITECTURE	18
2. WORKFLOW	18
3. USE CASE DIAGRAM	19
4. CLASS DIAGRAM	20
5. SEQUENCE DIAGRAM	20
6. ACTIVITY DIAGRAM	21
7. INSTALLATION OF PYTHON	41-45
8. BLOCKCHAIN	47
9. BLOCKCHAIN WORKS	48
10. HASH VALUE	51
11. TKINTER WIDGETS	53
12. PYFORM	56
13. PICKLE MODULE	57
14. RUN.BAT	63
15. SAVE CERTIFICATE WITH DIGITAL SIGNATURE	63
16. VERIFY CERTIFICATE	64
17. VERIFY CERTIFICATE WITHOUT DETAILS	64
18. CERTIFICATE VERIFIED	65
19. DIRECTLY VERIFYING CERTIFICATE	65
20. VERIFICATION FAILED	66

## **ABSTRACT**

In this project to secure academic certificate and for accurate management and to avoid forge certificate we are converting all certificates into digital signatures and this digital signature will be stored in Blockchain server as this Blockchain server support tamper proof data storage and nobody can hack or alter its data and if by an chance if its data alter then verification get failed at next block storage and user may get intimation about data alter.

In Blockchain technology same transaction data stored at multiple servers with hash code verification and if data alter at one server, then it will be detected from other server as for same data hash code will get different. For example, in Blockchain technology data will be stored at multiple servers and if malicious users alter data at one server, then its hash code will get changed in one server and other servers left unchanged and this changed hash code will be detected at verification time and future malicious user changes can be prevented.

In Blockchain each data will be stored by verifying old hash codes and if old hash codes remain unchanged then data will be considering as original and unchanged and then new transaction data will be appended to Blockchain as new block. For each new data storage all blocks hash code will be verified.

Keywords – Block chain, Hyperledger, Digital certificate, Hashing.

# 1. INTRODUCTION

## 1.1 Introduction:

Counterfeit academic certificates have been a longstanding issue in the academic community. Not until the Massachusetts Institute of Technology Media Lab released their project of Block-certs. A technique which is implemented by conflating the hash value of local files to the blockchain but remains numerous issues, did an effective technological approach protecting authentic credential certification and reputation appear.

Based on Block-certs, a series of cryptographic solutions are proposed to resolve the issues above, including, utilizing a multi-signature scheme to ameliorate the authentication of certificates. Exerting a safe revocation mechanism to improve the reliability of certificates revocation; establishing a secure federated identification to confirm the identity of the issuing institution. The project consists in designing and implementing the system which covered the above solutions. The project also involves a comprehensive evaluation of the system security. The assessment outcomes provide compelling evidence to prove that implementation is practical, reliable, secured, which might give some hints of important architectural considerations about the security attributes of other blockchain-based systems. In this section, we discuss the implementation from the point of view of system architecture, database architecture. The system architecture and database architecture show how the system is designed from the engineering point of view.

## 1.2 Overview:

The project consists in designing and implementing the system which covered the above solutions. The project also involves a comprehensive evaluation of the system security, and the assessment outcomes provide compelling evidence to prove that implementation is practical, reliable, secured, which might give some hints of important architectural considerations about the security attributes of other blockchain-based systems. In this section, we discuss the implementation from the point of view of system architecture, database architecture. The system architecture and database architecture show how the system is designed from the engineering point of view.

The issuing applications are responsible for the main business logic which include the certificates applying, examining, signing and issuing. The issuing applications are designed to merge the hash of the certificate in a Merkle tree and send the Merkle root to Blockchain amidst signing by the majority of community members. Also, the issuing applications involved the revocation of certificate. The issuing applications are responsible for the main business logic which includes the applying for, examining, signing and issuing of the certificates. The issuing applications are designed to merge the hash of the certificate with a Merkle tree and send the

Merkle root to the Blockchain. Also, the issuing applications deal with the revocations of certificates.

The verification application focuses on checking the authenticity and integrity of the certificates that have been issued. It includes two main components: a web-based page and an Android-based application. They use the same mechanism, and fetch the transaction message through the blockchain API and compare the transaction message with the verification data from the receipt. The mechanism can be briefly described in the following way: check the authentication code is valid; check the hash with the local certificate; confirm the hash is in the Merkle tree; ensure the Merkle root is in the blockchain; verify the certificate has not been revoked; validate the expired date of the certificate. Also, it has to be mentioned that for the convenience of sharing the certificates, the Android-based application allows for verification of the documents by scanning the QR code directly. The blockchain acts as the infrastructure of trust and a distributed database for saving the authentication data. Typically, the authentication data consist of the Merkle root generated using hashed data from thousands of certificates. The MongoDB is employed as our database since the MongoDB successfully manages JSON-based certificates and provides high availability and scalability.

Advances in information technology, the wide availability of the Internet, and common usage of mobile devices have changed the lifestyle of human beings. Virtual currency, digital coins originally designed for use online, has begun to be extensively adopted in real life. Because of the convenience of the Internet, various virtual currencies are thriving, including the most popular Bitcoin, Ether, and Ripple the value of which has surged recently. People are beginning to pay attention to blockchain, the backbone technology of these revolutionary currencies. Blockchain features a decentralized and incorruptible database that has high potential for a diverse range of uses

Blockchain is a distributed database that is widely used for recording distinct transactions. Once a consensus is reached among different nodes, the transaction is added to a block that already holds records of several transactions. Each block contains the hash value of its last counterpart for connection. All the blocks are connected and together they form a blockchain. Data are distributed among various nodes (the distributed data storage) and are thus decentralized. Consequently, the nodes maintain the database together. Under blockchain, a block becomes validated only once it has been verified by multiple

### **1.3 Problems or Issues being solved:**

- Certificate duplication can be minimized.
- Reduce the usage of paper.
- Easy to maintain the certificates or important documents.
- Users can get the information within fraction of seconds; time can be saved.
- By using blockchain fraud can be detected easily.

## **1.4 Objective:**

- To define a general schema and use cryptographic methods to create DigitalAcademic Certificates.
- To build a secure, scalable blockchain-based storage system for storage of the digital certificates.
- To develop an interface for data custodian and data consumer to store, manage, and deploy data certificates and upload, Authenticate certificates respectively.

## **1.5 Description:**

The physical certificates are converted into digital records. These digital records are stored in a blockchain network. Where the records once stored are immutable and cannot be changed forever due to the smart contracts. These digital records are created by the institution (issuer) and are deployed on the blockchain. A Certificate Id is sent to the user that is student or holder of the certificate. This Id can be sent to anyone intended to check or authenticate the digital certificate. For example, A company that is hiring the candidate and needs to verify the certificate authenticity of the candidate can verify it through our web application which is developed using MongoDB, React.JS, Node.JS, and Express.JS (MERN).

## **2.LITERATURE SURVEY**

### **2.1 LITERATURE SURVEY**

#### **2.1.1 Title: Blockchain as information Sharing with Personal data Store**

**Authors: Zibin Zheng**

**Description:** Provides a blockchain-based design to produce the credibility verification of the shared documents in period whereas maintaining necessary privacy. Discusses usage of blockchain to realize Associate in Nursing audit path of the accesses to the shared data. Whereas keeping the audit path non-public to the people involved.

#### **2.1.2 Title: Block-certs and the Digital Certificates**

**Authors: Richard Nuetey**

**Description:** An incubation project by the Media research laboratory Learning Initiative. Associate in Nursing the Learning Machine that builds a scheme for making, sharing, and validating bitcoin blockchain-based academic certificates. Digital certificates square measure registered on the Bitcoin blockchain, cryptographically signed, and tamper proof.

#### **2.1.3 Title: Blockchain and good Contract for Digital Certificate**

**Authors: A Gayatri**

**Description:** Jin-chiou developed a software package so as to avoid counterfeiting certificates. thanks to the dearth of Associate in Nursing anti-forge mechanism, the graduation certificate is to be solid. so, the decentralized application was designed to support Ethereum blockchain technology. First, generate the digital certificate for the paper certificate then hash worth created for the certificate is kept within the blockchain system. Even it wants to verify the credibility of the certificate it needed another scanning app to scan the certificate. The system saves on paper, stops document forgery. however, the QR Code should be scanned with a smartphone and an online association is needed.

#### **2.1.4 Title: Records Keeper**

**Authors: Omar S Saleh**

**Description:** Records Keeper is another blockchain-based mostly answer to verify educational certificates. With Records Keeper, academic institutes will issue certificates and supply a receipt to the user which might be shared with a 3rd party to prove the certificate is authentic. The receipt obtained from the scholar is employed by the third party to verify the certificate's credibility within the Record Keeper ledger. There don't seem to be several complications during this mechanism, however, the parties interested to look at the certificate within the Record Keeper blockchain should have possession rights. This amounts to a transfer of possession to the third party which can cause a change of state. this might work well on a non-public blockchain to make sure the safety of the certificate.

**3.1 Existing System:**

Smartcard is utilizing as a part of numerous routes like in web managing an account, internet business and shopping. Utilizing a similar strategy for keep up the authentications and reports. At whatever point the authentications are required they simply get to them by swiping the card. For this we are executing step 1 in the base paper i.e.; when we swipe the card the card ought to be approved to the savvy card, we are utilizing some calculation. In the wake of experiencing all the accessible calculations we have picked a calculation which can approve the card number. Calculation we have picked is LUHN calculation. LUHN calculation is likewise called as a modulo 10 calculation is a straightforward checksum recipe used to approve a recognizable proof number, for example, charge card numbers, IMEI numbers. The recipe confirms a number by including check digit, which is normally affixed to a halfway record number to create the full record number. The brilliant card number is approved by recipe. To create a paperless atmosphere without fraud using the well-used Smart card technology this can not only reduce the effort of maintaining the certificates but also used to create the technological innovation in the educational field with the help of ongoing trend.

**3.1.1 Disadvantages:**

- This process is time taken process, because every time we want to verify the certificate, the education authority must sign in and verify the certificate and generate the smart card to that certificate.
- In LUHAN algorithm, Learning takes long time. So automatically this impacted on execution speed also. i.e., low execution speed.
- Memory Requirement is high

**3.2 Proposed system:**

The main of this project to avoid tampering the certificates by using block chain. This very quick process to verify the certificates. Every certificate uploaded in the block chain has been assigned a hash code and digital sign to certificate. The certificate is verified through the hash code and digital sign. If the certificate has the same digital sign as previous certificate, then it displays the original certificate. In blockchain each data will be stored by verifying old hash codes and if old hash codes remain changed. Block Chain is a very high secure data storage, its store data in it and it cannot be modified or deleted. Based on Block-certs, a series of cryptographic solutions are proposed to resolve the issues above, including, utilizing a multi-signature scheme to ameliorate the authentication of certificates.

### **3.2.1 Advantages:**

- In Block Chain packages, Learning takes short time i.e., High execution speed.
- Comparison of certificates are fast i.e., high flexible.
- It has low memory requirements.

### **3.3 Feasibility:**

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

- ◆ ECONOMICAL FEASIBILITY
- ◆ TECHNICAL FEASIBILITY
- ◆ SOCIAL FEASIBILITY

#### **3.3.1 Economical Feasibility:**

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus, the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

#### **3.3.2 Technical Feasibility:**

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

#### **3.3.3 Social Feasibility:**

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.



## **4. SYSTEM REQUIREMENTS**

### **4.1 Software Requirements:**

The functional requirements or the overall description documents include the product perspective and features, operating system and operating environment, graphics requirements, design constraints and user documentation.

The appropriation of requirements and implementation constraints gives the general overview of the project regarding what the areas of strength and deficit are and how to tackle them.

- Operating system: Windows 7.
- Coding language: Python.
- IDE: Python IDLE.

### **4.2 Hardware Requirements:**

Minimum hardware requirements are very dependent on the particular software being developed by a given Enthought Python / Canopy / VS Code user. Applications that need to store large arrays/objects in memory will require more RAM, whereas applications that need to perform numerous calculations or tasks more quickly will require a faster processor.

- System: Intel Core i5 2.4 GHz.
- Hard Disk: As per required.
- RAM: 8GB.

### **4.3 Non-Functional Requirements:**

In systems engineering, a non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors. They are contrasted with functional requirements that define specific behavior or functions. The nonfunctional requirements can be considered as quality attributes of a system.

- Performance
- Reliability
- Efficiency
- Availability
- Maintainability
- Usability requirement
- Serviceability requirement
- Manageability requirement
- Recoverability requirement
- Security requirement

## **5. SYSTEM DESIGN**

### **5.1 UML DIAGRAMS:**

UML stands for Unified Modelling Language. UML is a standardized general-purpose modelling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object-oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modelling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modelling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems.

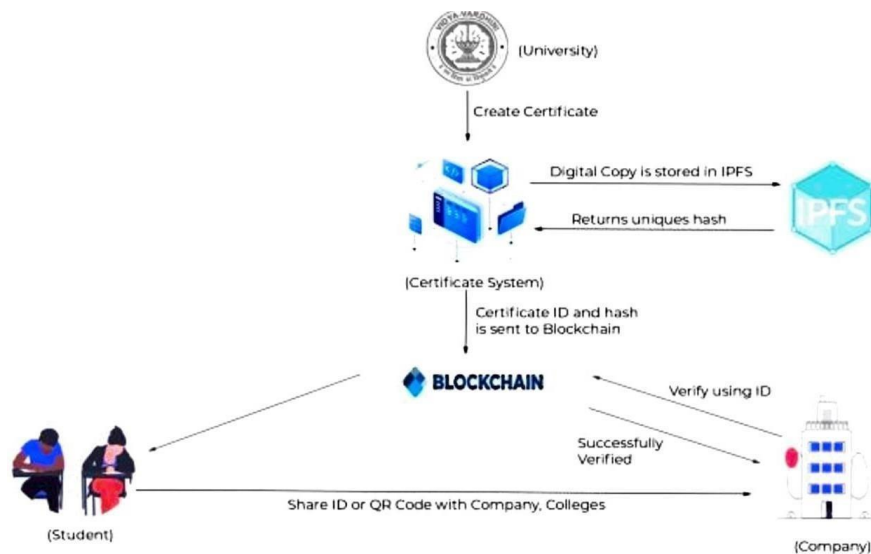
The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

### **GOALS:**

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modelling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modelling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

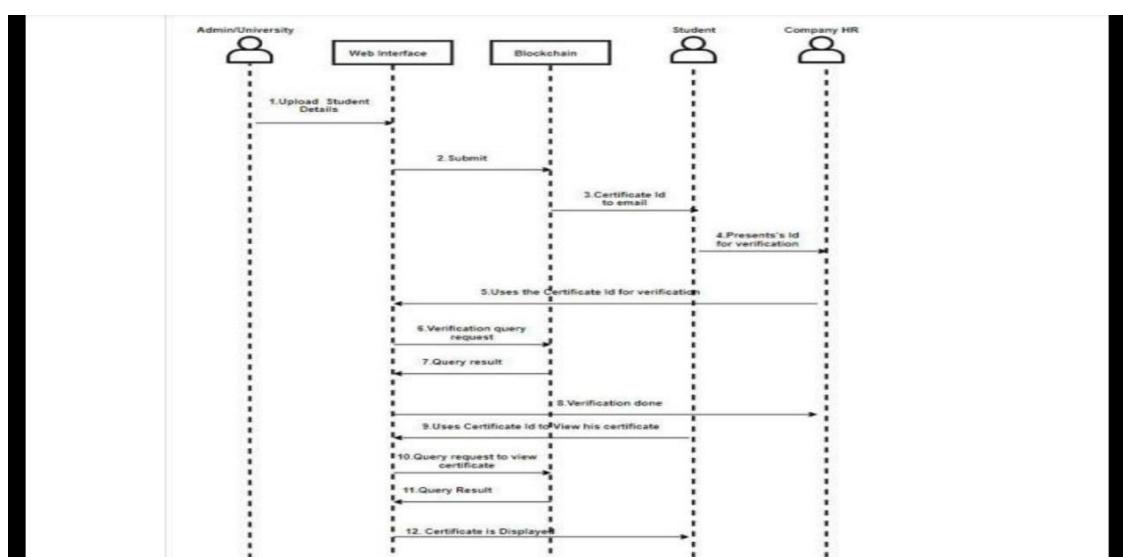
### 5.1.1 System Architecture:



**Fig 5.1.1 Architecture**

A model may be a complete, basic, and simplified description of software system design that consists of multiple views from a selected perspective or viewpoint as shown in figure 4.1. A read maybe an illustration of a complete system from the attitude of a connected set of issues. it's accustomed to describing the system from the point of view of various stakeholders like end-users, developers, project managers, and testers.

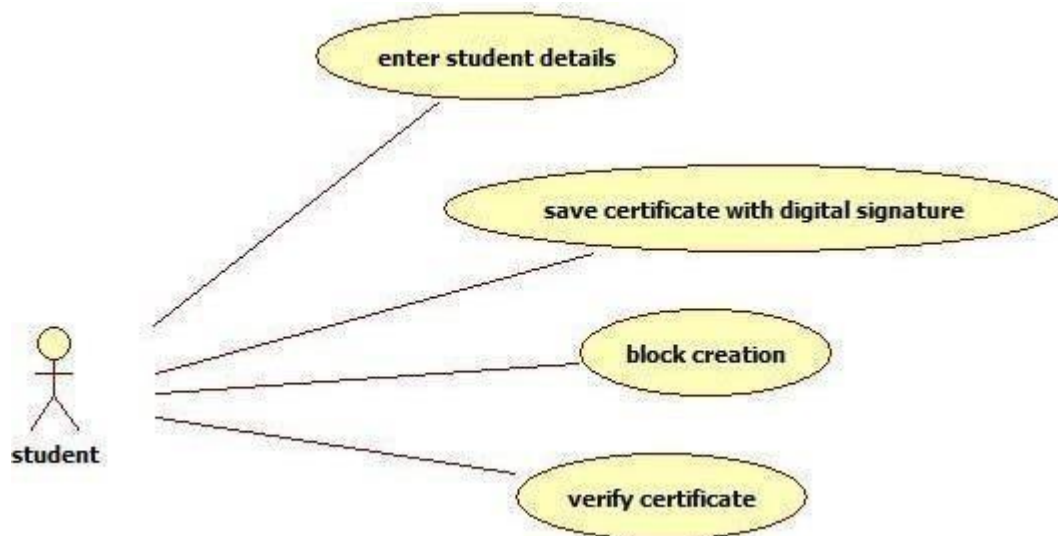
### 5.1.2 Workflow Diagram:



**Fig 5.1.2 Workflow Diagram**

Step 1: University will fill student academic details in a form provided using Dapp interface.  
Step 2: University submits the pre-viewed form details to Blockchain.  
Step 3: When the certificate is uploaded into Blockchain. Certificate hash code is generated and sent to respective student.  
Step 4: Students can share the Certificate hash code for certificate verification.  
Step 5: With the help of hash code Company HR can verify the certificate.  
Step 6: Verification of Certificate hash code query request will take place.  
Step 7: Query result is displayed i.e., verified or failed.  
Step 8: HR can confirm the certificate.  
Step 9: The student will use his Certificate hash code to view the certificate.  
Step 10: Query request to search Certificate hash code in the blockchain.  
Step 11: Query result for the Certificate hash code will be displayed.  
Step 12: Students can view their Certificate (Verification of certificate is optional for student).

### 5.1.3 Use Case Diagram:



**Fig 5.1.3: Use Case Diagram**

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

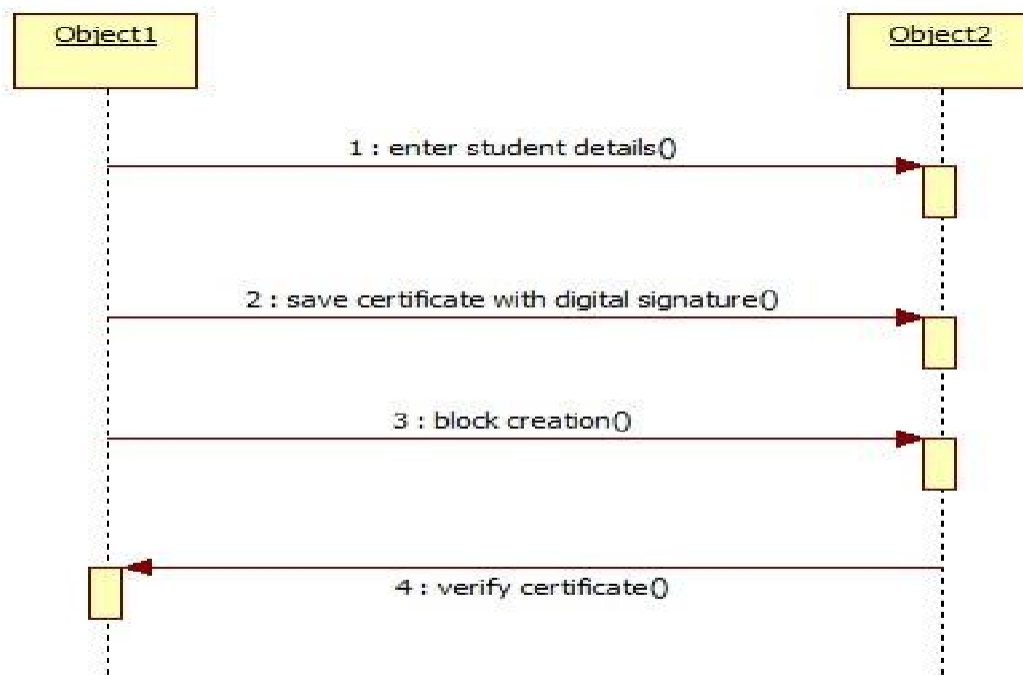
### 5.1.4 Class Diagram:



**Fig 5.1.4: Class Diagram**

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

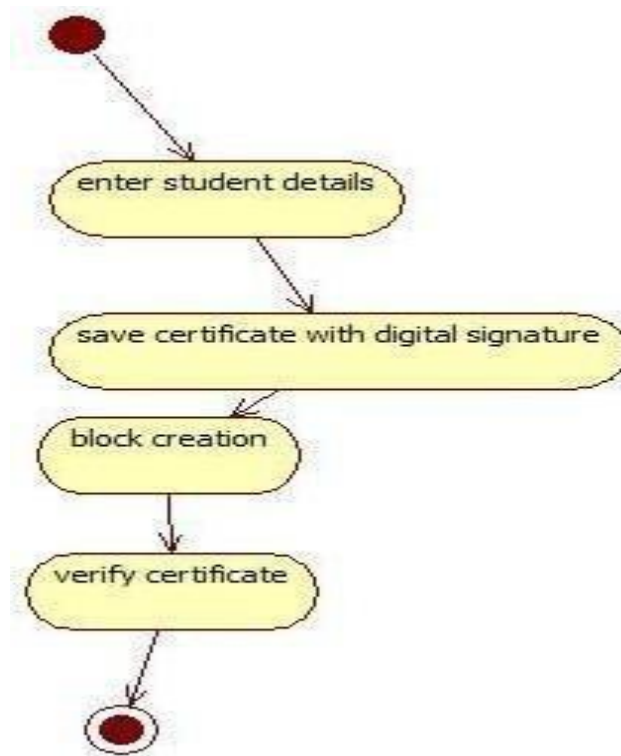
### 5.1.5 Sequence Diagram:



**Fig 5.1.5: Sequence Diagram**

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

### 5.1.6 Activity Diagram:



**Fig 5.1.6: Activity Diagram**

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

## **6. IMPLEMENTATION**

### **6.1 Module Description:**

The modules are required for effective purposes. They are,

#### **1) Save Certificate with Digital Signature:**

Using this module admin user can upload student details and student academic certificate and then application convert certificate into digital signature and then signature and other student details will be saved in Blockchain database.

#### **2) Verify Certificate:**

In this module verifier or companies or admin will take certificate from student and then upload to application and then application will convert certificate into digital signature and this digital signature will get checked/verified at Blockchain database and if matched found then Blockchain will retrieve all student details and display to verifier and if match not found then this certificate will be considered as fake or forge.

### **6.2 Algorithm:**

Step 1: Start

Step 2: Upload Datasets

Step 3: Enter student details

Step 4: Train Datasets

Step 5: Generate a hash value.

Step 6: Assign hash value to certificates.

Step 7: Verify certificate.

Step 8: Display valid or not.

### **6.3 Implementation Procedure:**

To create the block chain based unmodified certificates, initially the university must get registered. Each university getting to be added are going to be having its wallet address from which it's going to send transaction. University can be added only by the owner of the smart contract. Once added the university can access the system and may create certificates with data fields. Each created certificate is going to be stored within the Inter planetary filing system IPFS which successively will return the unique hash generated using SHA- 256 algorithm. This will function unique identity for every document. Along with this generated

hash and detail of certificates, all this data will be stored in the block chain. Anyone can use this transaction id to verify the certificate details and can view the original copy of certificate using IPFS hash stored along with data. And it's almost impossible to switch this certificate or to make fake certificate with same data. Hence with this we can solve the problem of counterfeit certificates.

Block chain is specialized for applications in data storage and make the secure. Block chain is mainly used to create a permanent, public, transparent ledger system for compiling data on sales, tracking digital use and payments to content creators. After the program initializes, it performs the certificate verification. If we upload a certificate give a hash code, digital sign to the certificate. If a new certificate is arrived, it should verify through previous certificate hash code and digital sign. If it matches it displays the real certificate. If the certificate is fake it provides the details of the original certificate at the display.



## 7. SYSTEM ENVIRONMENT

### 7.1 Python: -

Below are some facts about Python. Python is currently the most widely used multi-purpose, high-level programming language. Python allows programming in Object-Oriented and Procedural paradigms. Python programs generally are smaller than other programming languages like Java. Programmers have to type relatively less and indentation requirement of the language, makes them readable all the time. Python language is being used by almost all tech-giant companies like – Google, Amazon, Facebook, Instagram, Dropbox, Uber... etc. The biggest strength of Python is huge collection of standard libraries which can be used for the following –

- Blockchain
- Hash Values (also known as sha256)
- GUI Applications (like Tkinter, PyQt etc.)
- Pickle

#### 7.1.1 Advantages of Python:

Let's see how Python dominates over other languages.

##### 1. Extensive Libraries:

Python downloads with an extensive library and it contain code for various purposes like regular expressions, documentation-generation, unit-testing, web browsers, threading, databases, CGI, email, image manipulation, and more. So, we don't have to write the complete code for that manually.

##### 2. Extensible:

As we have seen earlier, Python can be extended to other languages. You can write some of your code in languages like C++ or C. This comes in handy, especially in projects.

##### 3. Embeddable:

Complimentary to extensibility, Python is embeddable as well. You can put your Python code in your source code of a different language, like C++. This lets us add scripting capabilities to our code in the other language.

##### 4. Improved Productivity:

The language's simplicity and extensive libraries render programmers more productive than languages like Java and C++ do. Also, the fact that you need to write less and get more things done.

##### 5. IOT Opportunities:

Since Python forms the basis of new platforms like Raspberry Pi, it finds the future bright for the Internet of Things. This is a way to connect the language with the real world.

## **6. Simple and Easy:**

When working with Java, you may have to create a class to print 'Hello World'. But in Python, just a print statement will do. It is also quite easy to learn, understand, and code. This is why when people pick up Python, they have a hard time adjusting to other more verbose languages like Java.

## **7. Readable:**

Because it is not such a verbose language, reading Python is much like reading English. This is the reason why it is so easy to learn, understand, and code. It also does not need curly braces to define blocks, and indentation is mandatory. This further aids the readability of the code.

## **8. Object-Oriented:**

This language supports both the procedural and object-oriented programming paradigms. While functions help us with code reusability, classes and objects let us model the real world. A class allows the encapsulation of data and functions into one.

## **9. Free and Open-Source:**

Like we said earlier, Python is freely available. But not only can you download Python for free, but you can also download its source code, make changes to it, and even distribute it. It downloads with an extensive collection of libraries to help you with your tasks.

## **10. Portable:**

When you code your project in a language like C++, you may need to make some changes to it if you want to run it on another platform. But it isn't the same with Python. Here, you need to code only once, and you can run it anywhere. This is called Write Once Run Anywhere (WORA). However, you need to be careful enough not to include any system-dependent features.

## **11. Interpreted:**

Lastly, we will say that it is an interpreted language. Since statements are executed one by one, debugging is easier than in compiled languages.

### **7.1.2 Advantages of Python Over Other Languages:**

#### **1. Less Coding:**

Almost all of the tasks done in Python requires less coding when the same task is done in other languages. Python also has an awesome standard library support, so you don't have to search for any third-party libraries to get your job done. This is the reason that many people suggest learning Python to beginners.

## **2. Affordable:**

Python is free therefore individuals, small companies or big organizations can leverage the free available resources to build applications. Python is popular and widely used so it gives you better community support.

## **3. Python is for Everyone:**

Python code can run on any machine whether it is Linux, Mac or Windows. Programmers need to learn different languages for different jobs but with Python, you can professionally build web apps, perform data analysis and machine learning, automate things, do web scraping and also build games and powerful visualizations. It is an all-rounder programming language.

### **7.1.3 Disadvantages of Python:**

So far, we've seen why Python is a great choice for your project. But if you choose it, you should be aware of its consequences as well. Let's now see the downsides of choosing Python over another language.

#### **1. Speed Limitations:**

We have seen that Python code is executed line by line. But since Python is interpreted, it often results in slow execution. This, however, isn't a problem unless speed is a focal point for the project. In other words, unless high speed is a requirement, the benefits offered by Python are enough to distract us from its speed limitations.

#### **2. Weak in Mobile Computing and Browsers:**

While it serves as an excellent server-side language, Python is much rarely seen on the client-side. Besides that, it is rarely ever used to implement smartphone-based applications. One such application is called Carbonnelle.

The reason it is not so famous despite the existence of Brython is that it isn't that secure.

#### **3. Design Restrictions:**

As you know, Python is dynamically-typed. This means that you don't need to declare the type of variable while writing the code. It uses duck-typing. But wait, what's that? Well, it just means that if it looks like a duck, it must be a duck. While this is easy on the programmers during coding, it can raise run-time errors.

#### **4. Underdeveloped Database Access Layers:**

Compared to more widely used technologies like JDBC (Java Database Connectivity) and ODBC (Open Database Connectivity), Python's database access layers are a bit underdeveloped. Consequently, it is less often applied in huge enterprises.

#### **5. Simple:**

No, we're not kidding. Python's simplicity can indeed be a problem. Take my example. I don't do Java, I'm more of a Python person. To me, its syntax is so simple that the verbosity of Java code seems unnecessary.

### **7.1.4 History of Python: -**

What do the alphabet and the programming language Python have in common? Right, both start with ABC. If we are talking about ABC in the Python context, it's clear that the programming language ABC is meant. ABC is a general-purpose programming language and programming environment, which had been developed in the Netherlands, Amsterdam, at the CWI (Centrum Wiskunde & Informatica). The greatest achievement of ABC was to influence the design of Python. Python was conceptualized in the late 1980s. Guido van Rossum worked that time in a project at the CWI, called Amoeba, a distributed operating system. In an interview with Bill Venners<sup>1</sup>, Guido van Rossum said: "In the early 1980s, I worked as an implementer on a team building a language called ABC at Centrum voor Wiskunde en Informatica (CWI). I don't know how well people know ABC's influence on Python. I try to mention ABC's influence because I'm indebted to everything I learned during that project and to the people who worked on it." Later on in the same Interview, Guido van Rossum continued: "I remembered all my experience and some of my frustration with ABC. I decided to try to design a simple scripting language that possessed some of ABC's better properties, but without its problems. So, I started typing. I created a simple virtual machine, a simple parser, and a simple runtime. I made my own version of the various ABC parts that I liked. I created a basic syntax, used indentation for statement grouping instead of curly braces or begin-end blocks, and developed a small number of powerful data types: a hash table (or dictionary, as we call it), a list, strings, and numbers."

## **What is Machine Learning : -**

Before we take a look at the details of various machine learning methods, let's start by looking at what machine learning is, and what it isn't. Machine learning is often categorized as a subfield of artificial intelligence, but I find that categorization can often be misleading at first brush. The study of machine learning certainly arose from research in this context, but in the data science application of machine learning methods, it's more helpful to think of machine learning as a means of building models of data.

Fundamentally, machine learning involves building mathematical models to help understand data. "Learning" enters the fray when we give these models tunable parameters that can be adapted to observed data; in this way the program can be considered to be "learning" from the data. Once these models have been fit to previously seen data, they can be used to predict and understand aspects of newly observed data. I'll leave to the reader the more philosophical digression regarding the extent to which this type of mathematical, model-based "learning" is similar to the "learning" exhibited by the human brain. Understanding the problem setting in machine learning is essential to using these tools effectively, and so we will start with some broad categorizations of the types of approaches we'll discuss here.

## **Categories Of Machine Learning: -**

At the most fundamental level, machine learning can be categorized into two main types: supervised learning and unsupervised learning.

Supervised learning involves somehow modeling the relationship between measured features of data and some label associated with the data; once this model is determined, it can be used to apply labels to new, unknown data. This is further subdivided into classification tasks and regression tasks: in classification, the labels are discrete categories, while in regression, the labels are continuous quantities. We will see examples of both types of supervised learning in the following section.

Unsupervised learning involves modeling the features of a dataset without reference to any label, and is often described as "letting the dataset speak for itself." These models include tasks such as clustering and dimensionality reduction. Clustering algorithms identify distinct groups of data, while dimensionality reduction algorithms search for more succinct

representations of the data. We will see examples of both types of unsupervised learning in the following section.

## **Need for Machine Learning**

Human beings, at this moment, are the most intelligent and advanced species on earth because they can think, evaluate and solve complex problems. On the other side, AI is still in its initial stage and haven't surpassed human intelligence in many aspects. Then the question is that what is the need to make machine learn? The most suitable reason for doing this is, "to make decisions, based on data, with efficiency and scale".

Lately, organizations are investing heavily in newer technologies like Artificial Intelligence, Machine Learning and Deep Learning to get the key information from data to perform several real-world tasks and solve problems. We can call it data-driven decisions taken by machines, particularly to automate the process. These data-driven decisions can be used, instead of using programming logic, in the problems that cannot be programmed inherently. The fact is that we can't do without human intelligence, but other aspect is that we all need to solve real-world problems with efficiency at a huge scale. That is why the need for machine learning arises.

## **Challenges in Machines Learning: -**

While Machine Learning is rapidly evolving, making significant strides with cybersecurity and autonomous cars, this segment of AI as whole still has a long way to go. The reason behind is that ML has not been able to overcome number of challenges. The challenges that ML is facing currently are –

**Quality of data** – Having good-quality data for ML algorithms is one of the biggest challenges. Use of low-quality data leads to the problems related to data preprocessing and feature extraction.

**Time-Consuming task** – Another challenge faced by ML models is the consumption of time especially for data acquisition, feature extraction and retrieval.

**Lack of specialist persons** – As ML technology is still in its infancy stage, availability of expert resources is a tough job.

**No clear objective for formulating business problems** – Having no clear objective and well-defined goal for business problems is another key challenge for ML because this technology is not that mature yet.

**Issue of overfitting & underfitting** – If the model is overfitting or underfitting, it cannot be represented well for the problem.

**Curse of dimensionality** – Another challenge ML model faces is too many features of data points. This can be a real hindrance.

**Difficulty in deployment** – Complexity of the ML model makes it quite difficult to be deployed in real life.

### **Applications of Machines Learning: -**

Machine Learning is the most rapidly growing technology and according to researchers we are in the golden year of AI and ML. It is used to solve many real-world complex problems which cannot be solved with traditional approach. Following are some real-world applications of ML –

- Emotion analysis
- Sentiment analysis
- Error detection and prevention
- Weather forecasting and prediction
- Stock market analysis and forecasting
- Speech synthesis
- Speech recognition
- Customer segmentation
- Object recognition
- Fraud detection
- Fraud prevention
- Recommendation of products to customer in online shopping

## How to Start Learning Machine Learning?

Arthur Samuel coined the term “**Machine Learning**” in 1959 and defined it as a “**Field of study that gives computers the capability to learn without being explicitly programmed**”.

And that was the beginning of Machine Learning! In modern times, Machine Learning is one of the most popular (if not the most!) career choices. According to Indeed, Machine Learning Engineer Is The Best Job of 2019 with a 344% growth and an average base salary of **\$146,085** per year.

But there is still a lot of doubt about what exactly is Machine Learning and how to start learning it? So this article deals with the Basics of Machine Learning and also the path you can follow to eventually become a full-fledged Machine Learning Engineer. Now let's get started!!!

### How to start learning ML?

This is a rough roadmap you can follow on your way to becoming an insanely talented Machine Learning Engineer. Of course, you can always modify the steps according to your needs to reach your desired end-goal!

#### Step 1 – Understand the Prerequisites

In case you are a genius, you could start ML directly but normally, there are some prerequisites that you need to know which include Linear Algebra, Multivariate Calculus, Statistics, and Python. And if you don't know these, never fear! You don't need a Ph.D. degree in these topics to get started but you do need a basic understanding.

##### (a) Learn Linear Algebra and Multivariate Calculus

Both Linear Algebra and Multivariate Calculus are important in Machine Learning. However, the extent to which you need them depends on your role as a data scientist. If you are more focused on application heavy machine learning, then you will not be that heavily focused on maths as there are many common libraries available. But if you want to focus



on R&D in Machine Learning, then mastery of Linear Algebra and Multivariate Calculus is very important as you will have to implement many ML algorithms from scratch.

#### (b) Learn Statistics

Data plays a huge role in Machine Learning. In fact, around 80% of your time as an ML expert will be spent collecting and cleaning data. And statistics is a field that handles the collection, analysis, and presentation of data. So it is no surprise that you need to learn it!!! Some of the key concepts in statistics that are important are Statistical Significance, Probability Distributions, Hypothesis Testing, Regression, etc. Also, Bayesian Thinking is also a very important part of ML which deals with various concepts like Conditional Probability, Priors, and Posteriors, Maximum Likelihood, etc.

#### (c) Learn Python

Some people prefer to skip Linear Algebra, Multivariate Calculus and Statistics and learn them as they go along with trial and error. But the one thing that you absolutely cannot skip is Python! While there are other languages you can use for Machine Learning like R, Scala, etc. Python is currently the most popular language for ML. In fact, there are many Python libraries that are specifically useful for Artificial Intelligence and Machine Learning such as Keras, TensorFlow, Scikit-learn, etc.

So if you want to learn ML, it's best if you learn Python! You can do that using various online resources and courses such as **Fork Python** available Free on GeeksforGeeks.

### Step 2 – Learn Various ML Concepts

Now that you are done with the prerequisites, you can move on to actually learning ML (Which is the fun part!!!) It's best to start with the basics and then move on to the more complicated stuff. Some of the basic concepts in ML are:

#### (a) Terminologies of Machine Learning

- **Model** – A model is a specific representation learned from data by applying some machine learning algorithm. A model is also called a hypothesis.

- **Feature** – A feature is an individual measurable property of the data. A set of numeric features can be conveniently described by a feature vector. Feature vectors are fed as input to the model. For example, in order to predict a fruit, there may be features like color, smell, taste, etc.
- **Target (Label)** – A target variable or label is the value to be predicted by our model. For the fruit example discussed in the feature section, the label with each set of input would be the name of the fruit like apple, orange, banana, etc.
- **Training** – The idea is to give a set of inputs(features) and it's expected outputs(labels), so after training, we will have a model (hypothesis) that will then map new data to one of the categories trained on.
- **Prediction** – Once our model is ready, it can be fed a set of inputs to which it will provide a predicted output(label).

### (b) Types of Machine Learning

- **Supervised Learning** – This involves learning from a training dataset with labeled data using classification and regression models. This learning process continues until the required level of performance is achieved.
- **Unsupervised Learning** – This involves using unlabelled data and then finding the underlying structure in the data in order to learn more and more about the data itself using factor and cluster analysis models.
- **Semi-supervised Learning** – This involves using unlabelled data like Unsupervised Learning with a small amount of labeled data. Using labeled data vastly increases the learning accuracy and is also more cost-effective than Supervised Learning.
- **Reinforcement Learning** – This involves learning optimal actions through trial and error. So the next action is decided by learning behaviors that are based on the current state and that will maximize the reward in the future.

### Advantages of Machine learning :-

1. Easily identifies trends and patterns-

Machine Learning can review large volumes of data and discover specific trends and patterns that would not be apparent to humans. For instance, for an e-commerce website like Amazon, it serves to understand the browsing behaviors and purchase histories of its users to help cater

to the right products, deals, and reminders relevant to them. It uses the results to reveal relevant advertisements to them.

## 2. No human intervention needed(automation)

With ML, you don't need to babysit your project every step of the way. Since it means giving machines the ability to learn, it lets them make predictions and also improve the algorithms on their own. A common example of this is anti-virus softwares; they learn to filter new threats as they are recognized. ML is also good at recognizing spam.

## 3. Continuous Improvement

As **ML algorithms** gain experience, they keep improving in accuracy and efficiency. This lets them make better decisions. Say you need to make a weather forecast model. As the amount of data you have keeps growing, your algorithms learn to make more accurate predictions faster.

## 4. Handling multi-dimensional and multi-variety data

Machine Learning algorithms are good at handling data that are multi-dimensional and multi-variety, and they can do this in dynamic or uncertain environments.

## 5. Wide Applications

You could be an e-tailer or a healthcare provider and make ML work for you. Where it does apply, it holds the capability to help deliver a much more personal experience to customers while also targeting the right customers.

## **Disadvantages of Machine Learning :-**

### 1. Data Acquisition

Machine Learning requires massive data sets to train on, and these should be inclusive/unbiased, and of good quality. There can also be times where they must wait for new data to be generated.

### 2. Time and Resources

ML needs enough time to let the algorithms learn and develop enough to fulfill their purpose with a considerable amount of accuracy and relevancy. It also needs massive resources to function. This can mean additional requirements of computer power for you.

### 3. Interpretation of Results

Another major challenge is the ability to accurately interpret results generated by the algorithms. You must also carefully choose the algorithms for your purpose.

### 4. High error-susceptibility

**Machine Learning** is autonomous but highly susceptible to errors. Suppose you train an algorithm with data sets small enough to not be inclusive. You end up with biased predictions coming from a biased training set. This leads to irrelevant advertisements being displayed to customers. In the case of ML, such blunders can set off a chain of errors that can go undetected for long periods of time. And when they do get noticed, it takes quite some time to recognize the source of the issue, and even longer to correct it.

### 7.1.5 Python Development Steps: -

Guido Van Rossum published the first version of Python code (version 0.9.0) at alt.sources in February 1991. This release included already exception handling, functions, and the core data types of list, dict, str and others. It was also object oriented and had a module system. Python version 1.0 was released in January 1994. The major new features included in this release were the functional programming tools lambda, map, filter and reduce, which Guido Van Rossum never liked. Six and a half years later in October 2000, Python 2.0 was introduced. This release included list comprehensions, a full garbage collector and it was supporting unicode. Python flourished for another 8 years in the versions 2.x before the next major release as Python 3.0 (also known as "Python 3000" and "Py3K") was released. Python 3 is not backwards compatible with Python 2.x. The emphasis in Python 3 had been on the removal of duplicate programming constructs and modules, thus fulfilling or coming close to fulfilling the 13th law of the Zen of Python: "There should be one -- and preferably only one -- obvious way to do it." Some changes in Python 7.3:

- Print is now a function
- Views and iterators instead of lists
- The rules for ordering comparisons have been simplified. E.g., a heterogeneous list cannot be sorted, because all the elements of a list must be comparable to each other.
- There is only one integer type left, i.e., int. long is int as well.

- The division of two integers returns a float instead of an integer. "/" can be used to have the "old" behavior.
- Text Vs. Data Instead of Unicode Vs. 8-bit

### **Purpose: -**

We demonstrated that our approach enables successful segmentation of intra-retinal layers—even with low-quality images containing speckle noise, low contrast, and different intensity ranges throughout—with the assistance of the ANIS feature.

### **Python**

Python is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace.

Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library.

- Python is Interpreted – Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- Python is Interactive – you can actually sit at a Python prompt and interact with the interpreter directly to write your programs.

Python also acknowledges that speed of development is important. Readable and terse code is part of this, and so is access to powerful constructs that avoid tedious repetition of code. Maintainability also ties into this may be an all but useless metric, but it does say something about how much code you have to scan, read and/or understand to troubleshoot problems or tweak behaviors. This speed of development, the ease with which a programmer of other languages can pick up basic Python skills and the huge standard library is key to another area where Python excels. All its tools have been quick to implement, saved a lot of time, and several of them have later been patched and updated by people with no Python background - without breaking.

## Modules Used in Project: -

### TensorFlow

TensorFlow is a free and open-source software library for dataflow and differentiable programming across a range of tasks. It is a symbolic math library, and is also used for machine learning applications such as neural networks. It is used for both research and production at Google.

TensorFlow was developed by the Google Brain team for internal Google use. It was released under the Apache 2.0 open-source license on November 9, 2015.

### NumPy

NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays. NumPy is one of the most commonly used packages for scientific computing in Python. It provides a multidimensional array object, as well as variations such as masks and matrices, which can be used for various math operations. NumPy is compatible with, and used by many other popular Python packages, including pandas and matplotlib. Quite simply, because it's faster than regular Python arrays, which lack numpy's optimized and pre-compiled C code that does all the heavy lifting. Another reason is that NumPy arrays and operations are vectorized, which means they lack explicit looping or indexing in the code. This makes the code not only more readable, but also more similar to standard mathematical notation.

It is the fundamental package for scientific computing with Python. It contains various features including these important ones:

- A powerful N-dimensional array object
- Sophisticated (broadcasting) functions
- Tools for integrating C/C++ and Fortran code
- Useful linear algebra, Fourier transform, and random number capabilities

Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined using NumPy which allows NumPy to seamlessly and speedily integrate with a wide variety of databases.

## **Pandas**

Pandas is an open-source Python Library providing high-performance data manipulation and analysis tool using its powerful data structures. Python was majorly used for data munging and preparation. It had very little contribution towards data analysis. Pandas solved this problem. Using Pandas, we can accomplish five typical steps in the processing and analysis of data, regardless of the origin of data load, prepare, manipulate, model, and analyze. Python with Pandas is used in a wide range of fields including academic and commercial domains including finance, economics, Statistics, analytics, etc.

Pandas is an open-source Python package that is most widely used for data science/data analysis and machine learning tasks. It is built on top of another package named NumPy, which provides support for multi-dimensional arrays. As one of the most popular data wrangling packages, Pandas works well with many other data science modules inside the Python ecosystem, and is typically included in every Python distribution, from those that come with your operating system to commercial vendor distributions like ActiveState's ActivePython.

### **Advantages**

- Fast and efficient for manipulating and analyzing data.
- Data from different file objects can be loaded.
- Easy handling of missing data (represented as NaN) in floating point as well as non-floating-point data
- Size mutability: columns can be inserted and deleted from Data Frame and higher dimensional objects
- Data set merging and joining.
- Flexible reshaping and pivoting of data sets
- Provides time-series functionality.
- Powerful group by functionality for performing split-apply-combine operations on data sets.

## Matplotlib

Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. Matplotlib can be used in Python scripts, the Python and IPython shells, the Jupyter Notebook, web application servers, and four graphical user interface toolkits. Matplotlib tries to make easy things easy and hard things possible. You can generate plots, histograms, power spectra, bar charts, error charts, scatter plots, etc., with just a few lines of code. For examples, see the sample plots and thumbnail gallery.

For simple plotting the pyplot module provides a MATLAB-like interface, particularly when combined with IPython. For the power user, you have full control of line styles, font properties, axes properties, etc, via an object-oriented interface or via a set of functions familiar to MATLAB users.

The pyplot API has a convenient MATLAB-style stateful interface. In fact, matplotlib was originally written as an open-source alternative for MATLAB. The OO API and its interface is more customizable and powerful than pyplot, but considered more difficult to use. As a result, the pyplot interface is more commonly used, and is referred to by default in this article.

Understanding matplotlib's pyplot API is key to understanding how to work with plots:

- matplotlib.pyplot.figure: Figure is the top-level container. It includes everything visualized in a plot including one or more Axes.
- matplotlib.pyplot.Axes: Axes contain most of the elements in a plot: *Axis*, *Tick*, *Line2D*, *Text*, etc., and sets the coordinates. It is the area in which data is plotted. Axes include the X-Axis, Y-Axis, and possibly a Z-Axis, as well.

Matplotlib is also available as uncompiled source files. Compiling from source will require your local system to have the appropriate compiler for your OS, all dependencies, setup scripts, configuration files, and patches available. This can result in a fairly complex installation. Alternatively, consider using the `ActiveState_Platform` to automatically build matplotlib from source and package it for your OS.

## Scikit – learn

Scikit-learn provides a range of supervised and unsupervised learning algorithms via a consistent interface in Python. It is licensed under a permissive simplified BSD license and is distributed under many Linux distributions, encouraging academic and commercial use. **Python**

Python is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace.

Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library.



- Python is Interpreted – Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- Python is Interactive – you can actually sit at a Python prompt and interact with the interpreter directly to write your programs.

Python also acknowledges that speed of development is important. Readable and terse code is part of this, and so is access to powerful constructs that avoid tedious repetition of code. Maintainability also ties into this may be an all but useless metric, but it does say something about how much code you have to scan, read and/or understand to troubleshoot problems or tweak behaviors. This speed of development, the ease with which a programmer of other languages can pick up basic Python skills and the huge standard library is key to another area where Python excels. All its tools have been quick to implement, saved a lot of time, and several of them have later been patched and updated by people with no Python background - without breaking.

### **Install Python Step-by-Step in Windows and Mac:**

Python a versatile programming language doesn't come pre-installed on your computer devices. Python was first released in the year 1991 and until today it is a very popular high-level programming language. Its style philosophy emphasizes code readability with its notable use of great whitespace.

The object-oriented approach and language construct provided by Python enables programmers to write both clear and logical code for projects. This software does not come pre-packaged with Windows.

#### **7.1.6 How to Install Python on Windows and Mac:**

There have been several updates in the Python version over the years. The question is how to install Python? It might be confusing for the beginner who is willing to start learning Python but this tutorial will solve your query. The latest or the newest version of Python is version 3.7.4 or in other words, it is Python 3.

**Note:** The python version 3.7.4 cannot be used on Windows XP or earlier devices.

Before you start with the installation process of Python. First, you need to know about your System Requirements. Based on your system type i.e., operating system and based processor, you must download the python version. My system type is a Windows 64-bit operating system. So, the steps below are to install python version 3.7.4 on Windows 7 device or to install Python 3. Download the Python Cheatsheet [here](#). The steps on how to install

Python on Windows 10, 8 and 7 are divided into 4 parts to help understand better.

Download the Correct version into the system

**Step 1:** Go to the official site to download and install python using Google Chrome or any other web browser. OR Click on the following link: <https://www.python.org>



Now, check for the latest and the correct version for your operating system.

**Step 2:** Click on the Download Tab.



**Step 3:** You can either select the Download Python for windows 3.7.4 button in Yellow Color or you can scroll further down and click on download with respective to their version. Here,

we are downloading the most recent python version for windows 3.7.4

Looking for a specific release?

Python releases by version number:

Release version	Release date		Click for more
<a href="#">Python 3.7.4</a>	July 8, 2019	<a href="#">Download</a>	<a href="#">Release Notes</a>
<a href="#">Python 3.6.9</a>	July 2, 2019	<a href="#">Download</a>	<a href="#">Release Notes</a>
<a href="#">Python 3.7.3</a>	March 25, 2019	<a href="#">Download</a>	<a href="#">Release Notes</a>
<a href="#">Python 3.4.10</a>	March 18, 2019	<a href="#">Download</a>	<a href="#">Release Notes</a>
<a href="#">Python 3.5.7</a>	March 18, 2019	<a href="#">Download</a>	<a href="#">Release Notes</a>
<a href="#">Python 2.7.16</a>	March 4, 2019	<a href="#">Download</a>	<a href="#">Release Notes</a>
<a href="#">Python 3.7.2</a>	Dec. 24, 2018	<a href="#">Download</a>	<a href="#">Release Notes</a>

**Step 4:** Scroll down the page until you find the Files option.

**Step 5:** Here you see a different version of python along with the operating system.

Files

Version	Operating System	Description	MD5 Sum	File Size	GPG
<a href="#">Unipped source tarball</a>	Source release		6811671a5b2db4ae77b9ab01b7079be	23017663	<a href="#">GPG</a>
<a href="#">XZ compressed source tarball</a>	Source release		d33e4aae6097051c3eca45ee3604803	17131432	<a href="#">GPG</a>
<a href="#">macOS 64-bit/32-bit installer</a>	Mac OS X	for Mac OS X 10.6 and later	6428b4fa7563daf1a442cbac0e08e6	34898416	<a href="#">GPG</a>
<a href="#">macOS 64-bit installer</a>	Mac OS X	for OS X 10.9 and later	5dd605c38217a45773b95ea936c2a3f	28882845	<a href="#">GPG</a>
<a href="#">Windows .hpg file</a>	Windows		06399573a2c96b2ac5aade0b4f7ed2	8131761	<a href="#">GPG</a>
<a href="#">Windows x86-64 embeddable zip file</a>	Windows	for AMD64/EM64T/x64	98093c9d8e9b0ab0c3134a40729a2	7504291	<a href="#">GPG</a>
<a href="#">Windows x86-64 executable installer</a>	Windows	for AMD64/EM64T/x64	a702b4b0ad76d9b0c3543a583e563400	26882348	<a href="#">GPG</a>
<a href="#">Windows x86-64 web-based installer</a>	Windows	for AMD64/EM64T/x64	28c31c908bd73ae8e51a3bd351b4bd2	1362904	<a href="#">GPG</a>
<a href="#">Windows x86 embeddable zip file</a>	Windows		9ab38d18b418795da9412574139d8	6741626	<a href="#">GPG</a>
<a href="#">Windows x86 executable installer</a>	Windows		33c8c2942d54446a3d8451476394789	25663848	<a href="#">GPG</a>
<a href="#">Windows x86 web-based installer</a>	Windows		1b670cfed117d03c30983ea371687c	1124608	<a href="#">GPG</a>

- To download Windows 32-bit python, you can select any one from the three options: Windows x86 embeddable zip file, Windows x86 executable installer or Windows x86 web-based installer.

- To download Windows 64-bit python, you can select any one from the three options: Windows x86-64 embeddable zip file, Windows x86-64 executable installer or Windows x86-64 web-based installer.

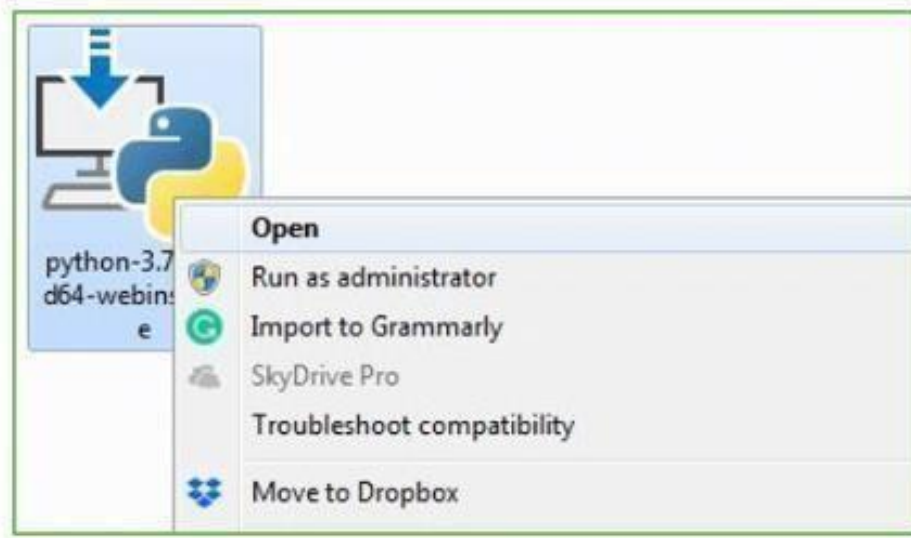
Here we will install Windows x86-64 web-based installer. Here your first part regarding

which version of python is to be downloaded is completed. Now we move ahead with the second part in installing python i.e., Installation

**Note:** To know the changes or updates that are made in the version you can click on the Release Note Option.

#### Installation of Python

**Step 1:** Go to Download and Open the downloaded python version to carry out the installation process.



**Step 2:** Before you click on Install Now, make sure to put a tick on Add Python 3.7 to PATH.



**Step 3:** Click on Install NOW After the installation is successful. Click on Close.



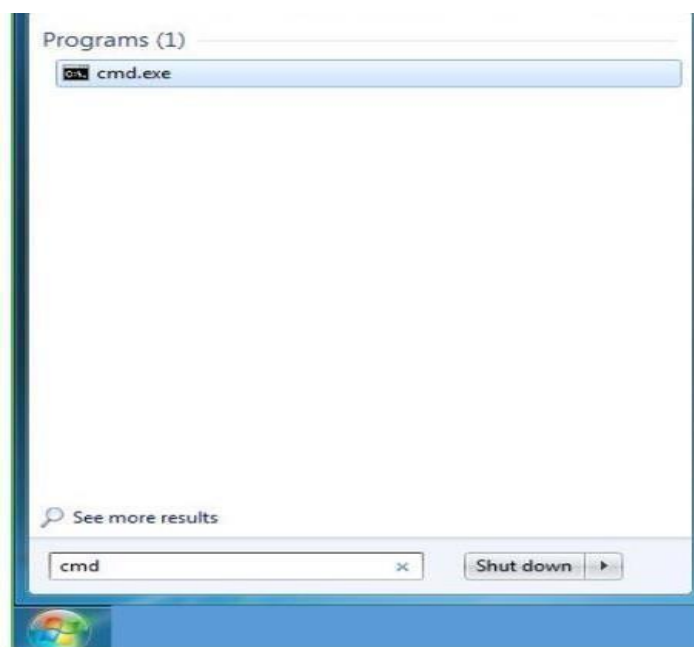
With these above three steps on python installation, you have successfully and correctly installed Python. Now is the time to verify the installation.

**Note:** The installation process might take a couple of minutes.

Verify the Python Installation

**Step 1:** Click on Start

**Step 2:** In the Windows Run Command, type “cmd”.



**Step 3:** Open the Command prompt option.

**Step 4:** Let us test whether the python is correctly installed. Type python -V and press Enter.



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\DELL>python -V
Python 3.7.4

C:\Users\DELL>_
```

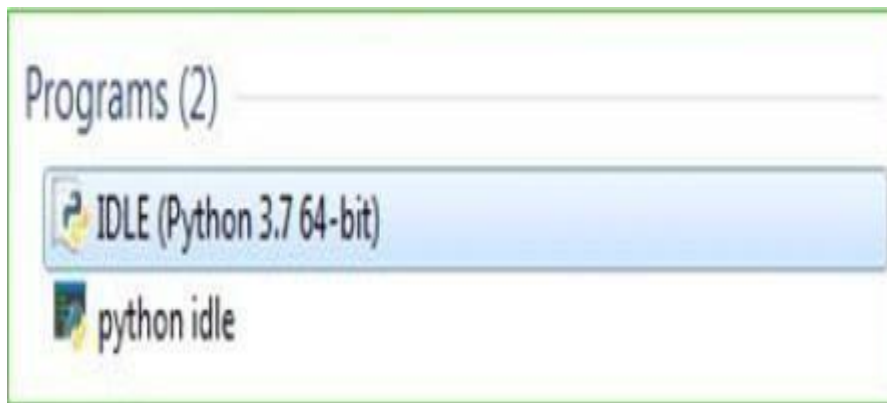
**Step 5:** You will get the answer as 3.7.4

**Note:** If you have any of the earlier versions of Python already installed. You must first uninstall the earlier version and then install the new one.

Check how the Python IDLE works

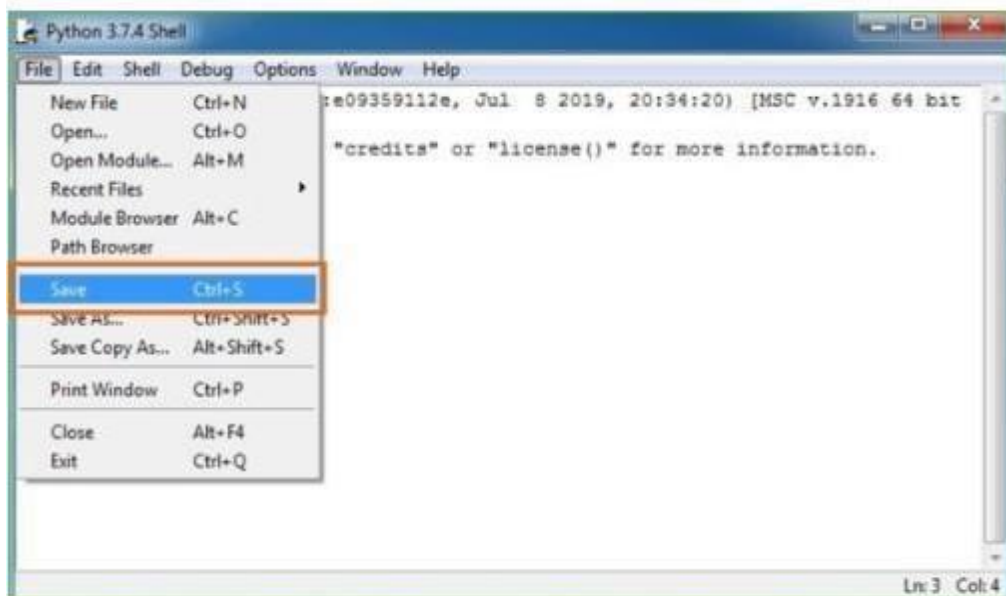
**Step 1:** Click on Start

**Step 2:** In the Windows Run command, type “python idle”.



**Step 3:** Click on IDLE (Python 3.7 64-bit) and launch the program

**Step 4:** To go ahead with working in IDLE you must first save the file. Click on File > Click on Save



**Step 5:** Name the file and save as type should be Python files. Click on SAVE. Here I have named the files as Hey World.

**Step 6:** Now for e.g., enter print

## 7.2 Blockchain

A blockchain is “a distributed database that maintains a continuously growing list of ordered records, called blocks.” These blocks “are linked using cryptography. Each block contains a cryptographic hash of the previous block, a timestamp, and transaction data. A blockchain is a decentralized, distributed, and public digital ledger that is used to record transactions across many computers so that the record cannot be altered retroactively without the alteration of all subsequent blocks and the consensus of the network.”

One key difference between a typical database and a blockchain is how the data is structured. A blockchain collects information together in groups, known as blocks, that hold sets of information. Blocks have certain storage capacities and, when filled, are closed and linked to the previously filled block, forming a chain of data known as the blockchain. All new information that follows that freshly added block is compiled into a newly formed block that will then also be added to the chain once filled.

A database usually structures its data into tables, whereas a blockchain, as its name implies, structures its data into chunks (blocks) that are strung together. This data structure inherently makes an irreversible timeline of data when implemented in a decentralized nature. When a block is filled, it is set in stone and becomes a part of this timeline. Each block in the chain is given an exact timestamp when it is added to the chain.



## The Properties of Distributed Ledger Technology (DLT)

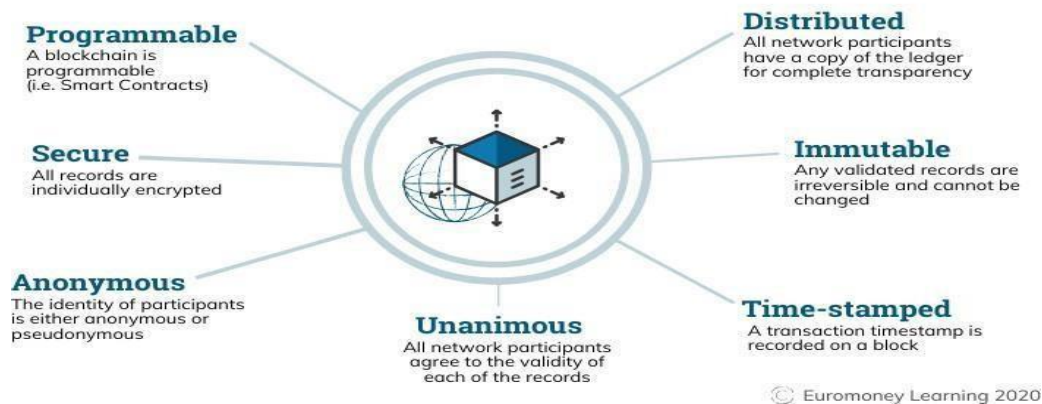


Fig 7.2Blockchain

### 7.2.1 Blockchain Works

Each block contains a hash (a digital fingerprint or unique identifier), timestamped batches of recent valid transactions, and the hash of the previous block. The previous block hash links the blocks together and prevents any block from being altered or a block being inserted between two existing blocks.” In theory, the method renders the blockchain tamperproof.

The two main types of blockchain, public and private, offer different levels of security. Public blockchains “use computers connected to the public internet to validate transactions and bundle them into blocks to add to the ledger. Private blockchains, on the other hand, typically only permit known organizations to join.” Because any organization can join public blockchains, they might not be right for enterprises concerned about the confidentiality of the information moving through the network.

Blockchain technologies are growing at an unprecedented rate and powering new concepts for everything from shared storage to social networks. From a security perspective, we are breaking new ground. As developers create blockchain applications, they should give precedent to securing their blockchain applications and services. Activities such as performing, creating threat models, and doing code analysis, security and should all be on a developer’s blockchain application roadmap. Building security in from the start is critical to ensuring a successful and secure blockchain application.

Blockchain is a type of shared database that differs from a typical database in the way that it stores information; blockchains store data in blocks that are then linked together via cryptography.

As new data comes in, it is entered into a fresh block. Once the block is filled with data, it is chained onto the previous block, which makes the data chained together in chronological order.

Different types of information can be stored on a blockchain, but the most common use so far has been as a ledger for transactions.



In Bitcoin's case, blockchain is used in a decentralized way so that no single person or group has control—rather, all users collectively retain control.

Decentralized blockchains are immutable, which means that the data entered is irreversible. For Bitcoin, this means that transactions are permanently recorded and viewable to anyone.

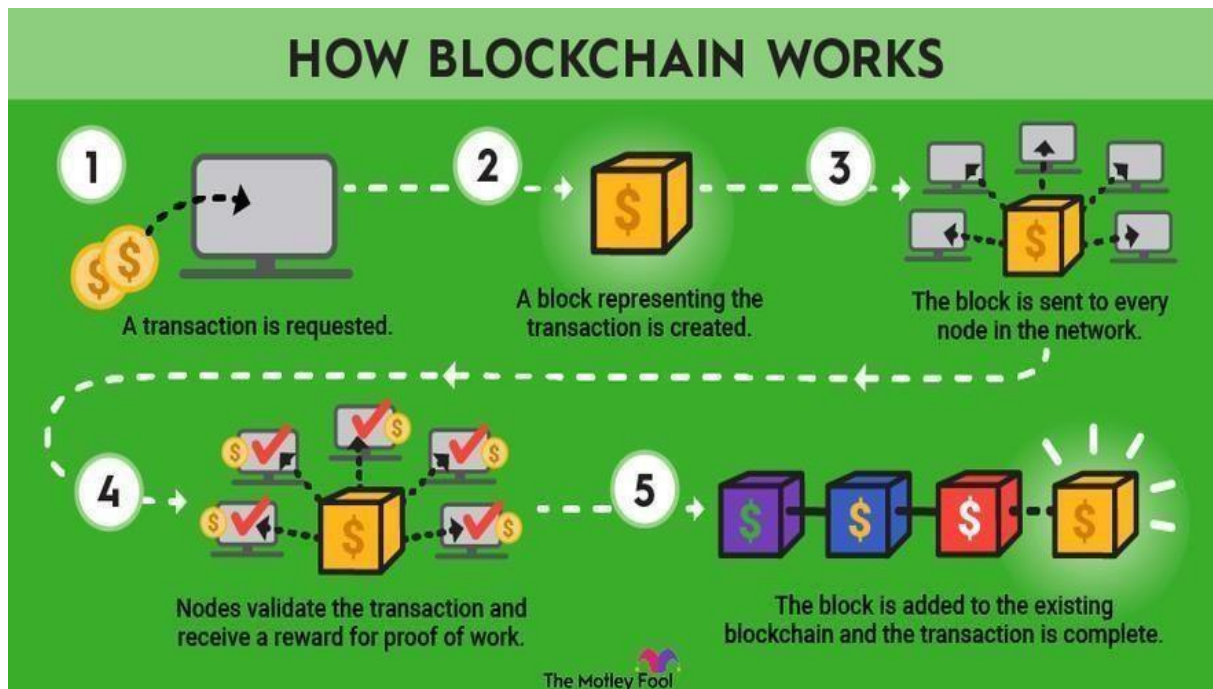


Fig 7.2.2 Blockchain Works

## 7.2.2 Building trust with Blockchain:

Blockchain enhances trust across a business network. It's not that you just can't trust those with who you conduct business with it's that you just don't get to once in operation on a Blockchain network. Blockchain builds trust through the subsequent for the attributes for this purpose.

- **Distributed:** The distributed ledger is shared and updated with each incoming group action among the nodes connected to the Blockchain. All this can be worn out real time as there's no central server dominant the information.

- **Secure:** there's no unauthorized access to Blockchain created attainable through Permissions and Cryptography.

- **Transparent:** as a result of each node or participant in Blockchain contains a copy of the Blockchain information, they need access to all or any group action information. They themselves will verify the identities while not the requirement for mediators.

- **Consensus-based:** All relevant network participants should agree that a group action is valid. this can be achieved through the utilization of agreement algorithms.

- **Flexible:** good Contracts that are dead supported bound conditions may be written into the platform. Blockchain networks will evolve in pace with business processes and also for the future.

## **7.2.3 Benefits of Blockchains**

### **Accuracy of the Chain**

Transactions on the blockchain network are approved by a network of thousands of computers. This removes almost all human involvement in the verification process, resulting in less human error and an accurate record of information. Even if a computer on the network were to make a computational mistake, the error would only be made to one copy of the blockchain. For that error to spread to the rest of the blockchain, it would need to be made by at least 51% of the network's computers—a near impossibility for a large and growing network the size of Bitcoin's.

6

### **Cost Reductions**

Typically, consumers pay a bank to verify a transaction, a notary to sign a document, or a minister to perform a marriage. Blockchain eliminates the need for third-party verification—and, with it, their associated costs. For example, business owners incur a small fee whenever they accept payments using credit cards, because banks and payment-processing companies have to process those transactions. Bitcoin, on the other hand, does not have a central authority and has limited transaction fees.

### **Decentralization**

Blockchain does not store any of its information in a central location. Instead, the blockchain is copied and spread across a network of computers. Whenever a new block is added to the blockchain, every computer on the network updates its blockchain to reflect the change. By spreading that information across a network, rather than storing it in one central database, blockchain becomes more difficult to tamper with. If a copy of the blockchain fell into the hands of a hacker, only a single copy of the information, rather than the entire network, would be compromised.

### **Efficient Transactions**

Transactions placed through a central authority can take up to a few days to settle. If you attempt to deposit a check on Friday evening, for example, you may not actually see funds in your account until Monday morning. Whereas financial institutions operate during business hours, usually five days a week, blockchain is working 24 hours a day, seven days a week, and 365 days a year. Transactions can be completed in as little as 10 minutes and can be considered secure after just a few hours. This is particularly useful for cross-border trades, which usually take much longer because of time zone issues and the fact that all parties must confirm payment processing.

### **Private Transactions**

Many blockchain networks operate as public databases, meaning that anyone with an Internet connection can view a list of the network's transaction history. Although users can access details about transactions, they cannot access identifying information about the users making those transactions. It is a common misperception that blockchain networks like bitcoin are anonymous, when in fact they are only confidential.

When a user makes a public transaction, their unique code—called a public key, as mentioned earlier—is recorded on the blockchain. Their personal information is not. If a person has made a Bitcoin purchase on an exchange that requires identification, then the person's identity is still linked to their blockchain address—but a transaction, even when tied to a person's name, does not reveal any personal information.

## **Secure Transactions**

Once a transaction is recorded, its authenticity must be verified by the blockchain network. Thousands of computers on the blockchain rush to confirm that the details of the purchase are correct. After a computer has validated the transaction, it is added to the blockchain block. Each block on the blockchain contains its own unique hash, along with the unique hash of the block before it. When the information on a block is edited in any way, that block's hash code changes—however, the hash code on the block after it would not. This discrepancy makes it extremely difficult for information on the blockchain to be changed without notice.

## **Transparency**

Most blockchains are entirely open-source software. This means that anyone and everyone can view its code. This gives auditors the ability to review cryptocurrencies like Bitcoin for security. This also means that there is no real authority on who controls Bitcoin's code or how it is edited. Because of this, anyone can suggest changes or upgrades to the system. If a majority of the network users agree that the new version of the code with the upgrade is sound and worthwhile, then Bitcoin can be updated.

## **Banking the Unbanked**

Perhaps the most profound facet of blockchain and Bitcoin is the ability for anyone, regardless of ethnicity, gender, or cultural background, to use it. According to The World Bank, an estimated 1.7 billion adults do not have bank accounts or any means of storing their money or wealth.

7

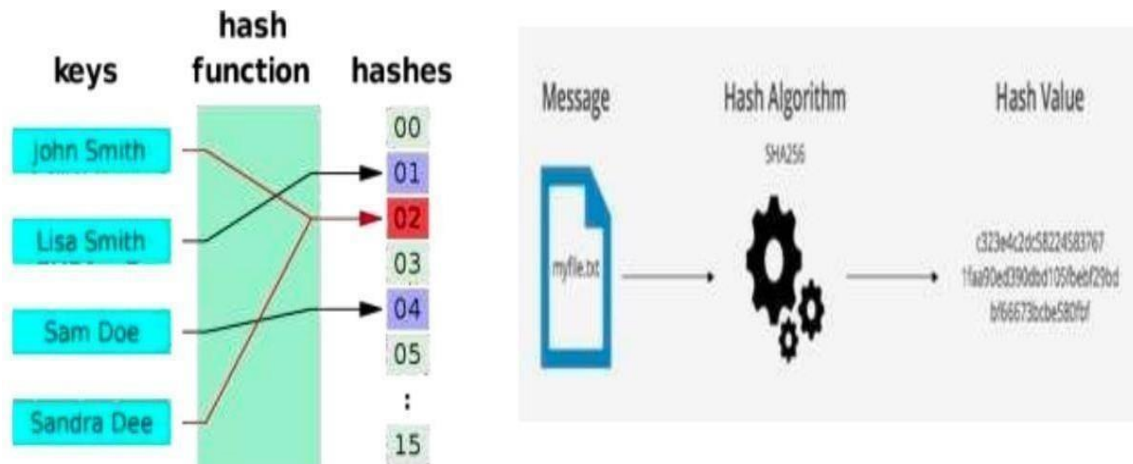
Nearly all of these individuals live in developing countries, where the economy is in its infancy and entirely dependent on cash.

These people often earn a little money that is paid in physical cash. They then need to store this physical cash in hidden locations in their homes or other places of living, leaving them subject to robbery or unnecessary violence. Keys to a bitcoin wallet can be stored on a piece of paper, a cheap cell phone, or even memorized if necessary. For most people, it is likely that these options are more easily hidden than a small pile of cash under a mattress.

Blockchains of the future are also looking for solutions to not only be a unit of account for wealth storage but also to store medical records, property rights, and a variety of other legal contracts.

## **7.3 Hash Value**

A hash function turns an input (for example text) into a string of bytes with a fixed length and structure. The output or value created is called a 'hash value' or 'checksum.' Any hash value created from data using a specific hashing algorithm is always the same length and one-way - it cannot be reversed.



**Fig 7.3 Hash value**

## 7.4 GUI Application and Types

There are many graphical user interface (GUI) toolkits that you can use with the Python programming language. The big three are Tkinter, python, and PyQt. Each of these toolkits will work with Windows, macOS, and Linux, with PyQt having the additional capability of working on mobile.

A graphical user interface is an application that has buttons, windows, and lots of other widgets that the user can use to interact with your application. A good example would be a web browser. It has buttons, tabs, and a main window where all the content loads.

- PyQt5
- Tkinter
- Kivy
- PySimpleGUI
- Libavg
- PyForms
- PySide2
- Wax

### 7.4.1 Tkinter

Tkinter is the standard GUI library for Python. Python when combined with Tkinter provides a fast and easy way to create GUI applications. Tkinter provides a powerful object-oriented interface to the Tk GUI toolkit.

Creating a GUI application using Tkinter is an easy task. All you need to do perform the following steps –

- Import the Tkinter module.
- Create the GUI application main window.
- Add one or more of the above-mentioned widgets to the GUI application.
- Enter the main event loop to take action against each event triggered by the user.

## Important Tk Concepts

Even this simple program illustrates the following key Tk concepts:

### Widgets:

A Tkinter user interface is made up of individual *widgets*. Each widget is represented as a Python object, instantiated from classes like `tk.Frame`, `tk.Label`, and `tk.Button`.

### widget hierarchy:

Widgets are arranged in a *hierarchy*. The label and button were contained within a frame, which in turn was contained within the root window. When creating each *child* widget, its *parent* widget is passed as the first argument to the widget constructor.

### configuration options:

Widgets have *configuration options*, which modify their appearance and behavior, such as the text to display in a label or button. Different classes of widgets will have different sets of options.

### geometry management:

Widgets aren't automatically added to the user interface when they are created. A *geometry manager* like `grid` controls where in the user interface they are placed.

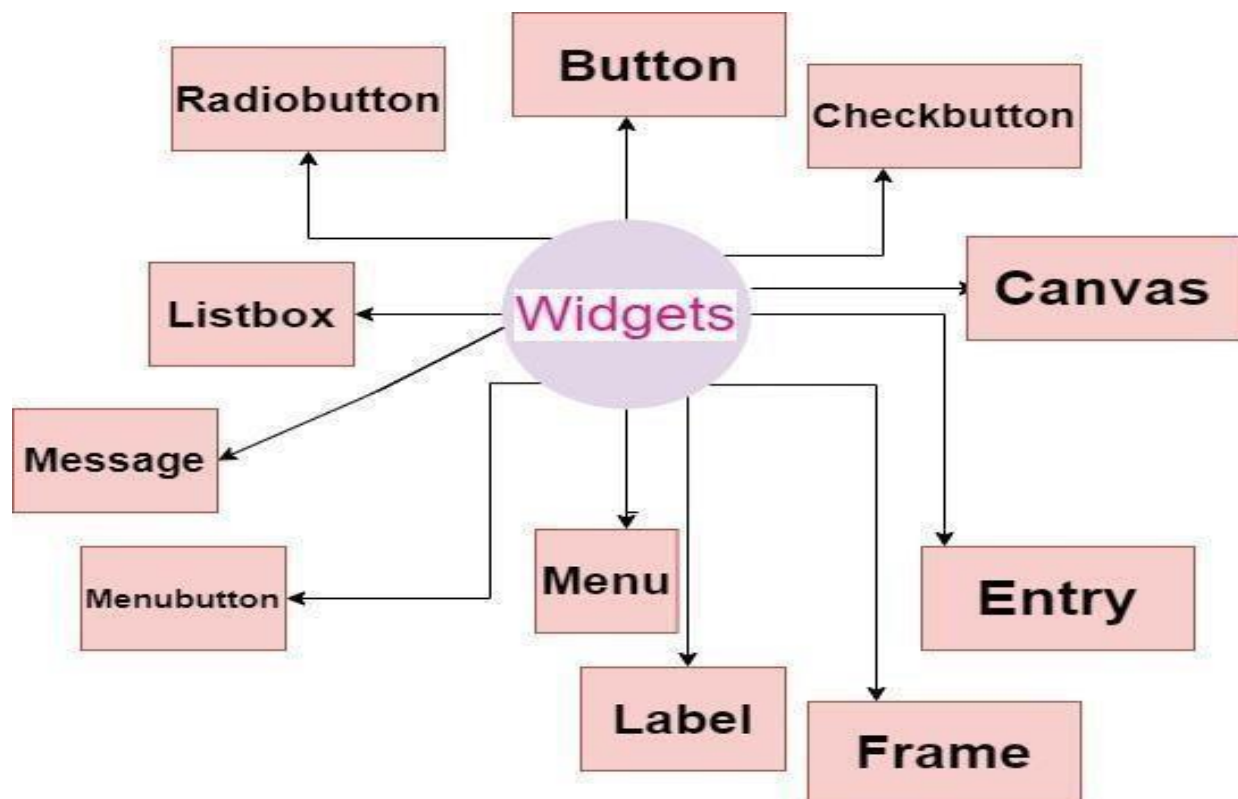
### event loop:

Tkinter reacts to user input, changes from your program, and even refreshes the display only when actively running an *event loop*. If your program isn't running the event loop, your user interface won't update.

## 7.4.2 Tkinter Widgets

Tkinter provides various controls, such as buttons, labels and text boxes used in a GUI application. These controls are commonly called widgets.

There are currently 15 types of widgets in Tkinter. We present these widgets as well as a brief description in the following table –



**Fig 7.4.2: Tkinter Widgets**

### 7.4.3 PyQt:

PyQt5 is the latest version of a GUI widgets toolkit developed by Riverbank Computing. It is a Python interface for Qt, one of the most powerful, and popular cross-platform GUI libraries. PyQt5 is a blend of Python programming language and the Qt library. This introductory tutorial will assist you in creating graphical applications with the help of PyQt.

There are so many options provided by Python to develop GUI application and PyQt5 is one of them. PyQt5 is cross-platform GUI toolkit, a set of python bindings for Qt v5. One can develop an interactive desktop application with so much ease because of the tools and simplicity provided by this library. A GUI application consists of Front-end and Back-end. PyQt5 has provided a tool called 'QtDesigner' to design the front-end by drag and drop method so that development can become faster and one can give more time on back-end stuff. Installation. First, we need to install PyQt5 library.

PyQt is one of the most popular Python bindings for the Qt cross-platform C++ framework. PyQt was developed by Riverbank Computing Limited. Qt itself is developed as part of the Qt Project. PyQt provides bindings for Qt 4 and Qt 5. PyQt is distributed under a choice of licences: GPL version 3 or a commercial license.

PyQt is available in two editions: PyQt4 which will build against Qt 4.x and 5.x and PyQt5 which will only build against 5.x. Both editions can be built for Python 2 and 3. PyQt contains over 620 classes that cover graphical user interfaces, XML handling, network communication, SQL databases, Web browsing and other technologies available in Qt. The latest iteration of PyQt is v5.11.3. It fully supports Qt 5.11.2. PyQt4 runs on Windows, Linux, Mac OS X and various UNIX platforms. PyQt5 also runs on Android and iOS.



## 7.4.4 Kivy

Kivy tutorial provides basic and advanced concepts of Kivy. Our Kivy tutorial is designed for beginners as well as working professionals. Kivy is an open-source and Graphical User Interface (GUI) development platform for Python. It helps us to develop mobile applications and multi-touch application software with a NUI (Natural User Interface). It allows developers to build an application once and use it across all devices. It can also access mobile APIs to manipulate things such as the camera on a phone, GPS tracking, vibrator, and so on. It contains various components for creating an application such as: A graphical library OpenGL ES 2. An extensive range of widgets that support multi-touch. An intermediate Kv language to design custom widgets. An extensive support for input devices such as a mouse, keyboard, TUIO, as well as OS-specific multi-touch events. Kivy is a multi-platform application development framework for Python. It allows us to develop multi-platform applications on various platforms such as Windows, Linux, Android, macOS, iOS, and Raspberry Pi.

Kivy is an open-source Python library that allows you to develop multi-platform graphical user interface applications on Windows, macOS, Android, iOS, Linux, and Raspberry-Pi. In addition to regular mouse and keyboard inputs, it supports multitouch events. Applications made using Kivy will appear similar across all the platforms but it also means that the applications feel or look will differ from any native application.

## 7.4.5 PySimple

The PySimpleGui project started as a wrapper around TKinter package, which is bundled with Python's standard library, with the objective to simplify the GUI building process. PySimpleGui subsequently added the ability to design desktop GUIs based on PySide library (which itself ports Qt GUI toolkit, originally written in C++, to Python) and Python (which ports another popular GUI toolkit called WxWidgets). These libraries are called PySimpleGUIQt and PySimpleGUIWx respectively. The latest addition to the PySimpleGui family is the PySimpleGUIWeb package which uses the Remi (Remote Interface Library) to construct GUI design that is rendered in a web page. All the packages in the PySimpleGui group follow the similar API, which means the names of GUI elements, their properties and methods are same in all the four packages. As a result, just by replacing the import statement (and keeping the rest of the code unchanged), one can get the corresponding GUI design rendered. This is in fact the most important feature of PySimpleGui. That's why, it is known as Python GUIs for Humans.

PySimpleGUI is a Python package that enables Python programmers of all levels to create GUIs. You specify your GUI window using a "layout" which contains widgets (they're called "Elements" in PySimpleGUI). Your layout is used to create a window using one of the 4 supported frameworks to display and interact with your window. Supported frameworks include tkinter, Qt, Wx Python, or Remi. The term "wrapper" is sometimes used for these kinds of packages.

Your PySimpleGUI code is simpler and shorter than writing directly using the underlying framework because PySimpleGUI implements much of the "boilerplate code" for you. Additionally, interfaces are simplified to require as little code as possible to get the desired result. Depending on the program and framework used, a PySimpleGUI program may require 1/2 to 1/10th amount of code to create an identical window using one of the frameworks directly.

While the goal is to encapsulate/hide the specific objects and code used by the GUI framework you are running on top of, if needed you can access the frameworks' dependent widgets and windows directly. If a setting or feature is not yet exposed or accessible using the PySimpleGUI APIs, you are not walled off from the framework. You can expand capabilities without directly modifying the PySimpleGUI package itself.

### **7.4.6 Libavg**

Libavg is great for development of modern touch UIs. It supports all major touch driver models, including Windows touch, Linux XInput, and TUIO. Just as importantly, it has a full-featured event handling system that includes support for direct manipulation and gesture recognition. Libavg allows programmers, media artists and designers to quickly develop media applications. It uses python as scripting language, is written in high-speed C++ and uses modern OpenGL for display output. The project has been under constant development since its inception in 2003.

Libavg supports the full variety of display elements - images, text, videos, camera output, vector graphics - that modern graphics-intensive applications need. It is fast: the layout engine supports thousands of display elements on the screen at once as well as hardware-accelerated video output. Text using markup, videos with an alpha channel, rendering to offscreen buffers, masking, as well as GPU shader effects such as blur and chromakey are all supported. Plugins written in C++ can expand the engine and have access to all libavg internals. The Tutorial explains all these concepts using a sample application. libavg runs on Linux, Mac OS X and Windows, is open source and licensed under the LGPL. Development is sponsored by Archimedes Exhibitions (which uses libavg for its exhibits) and by the Interactive Media Lab Dresden (where it is being used for research and education).

Libavg packages available for several system configurations. This page describes how to install them. If you don't find your configuration here, if you want the newest developer version, or if you'd like to compile libavg yourself for some other reason, have a look at one of the following pages:

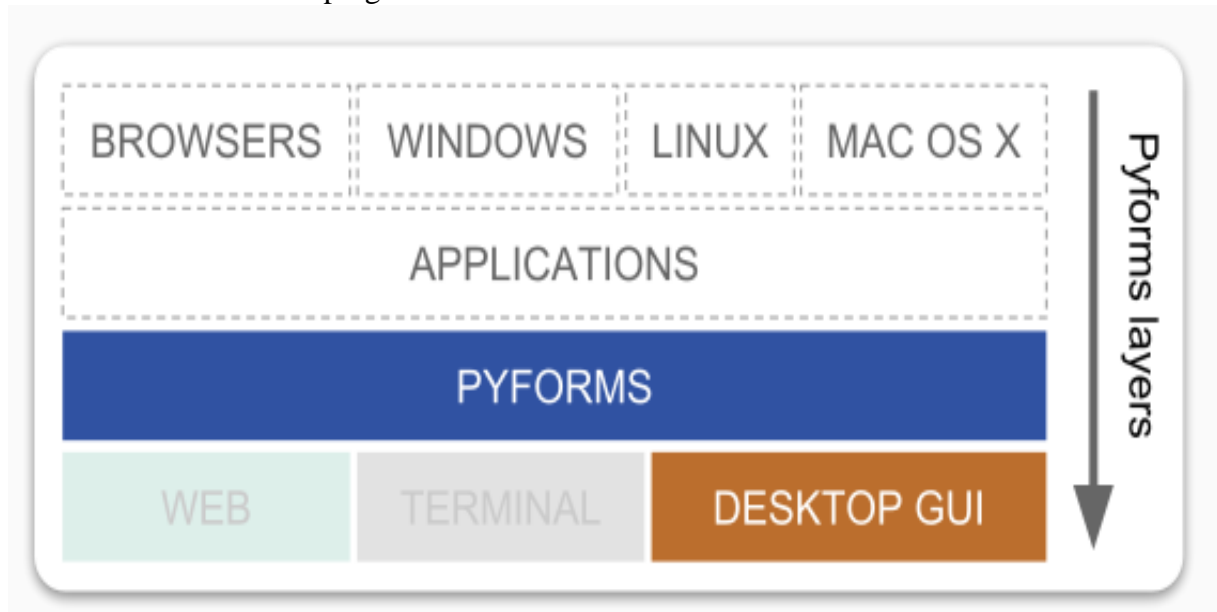
- [MacSourceInstall](#)
- [UbuntuSourceInstall](#)
- [WinSourceInstall](#)
- [RaspberryPISourceInstall](#)

### **7.4.7 PyForms**

Pyforms is a Python 3 cross-environment framework that aims to boost the development productivity. The library provides an API in Python to develop applications that can be executed in Windows GUI mode, Web mode, or in Terminal mode. Pyforms is a Python 3 cross-environment framework that aims to boost the development productivity. The library provides an API in Python to develop applications that can be executed in Windows GUI mode, Web mode, or in Terminal mode. After looking into the several python options for GUI interfaces, PyQt was the one that seemed the best tool for a fast development with the Designer, but after a while developing in Qt, switching between the designer and the python IDE was becoming too costly in terms of time because the interfaces were constantly evolving.



Being a Django developer, I did get inspiration on it for this framework. In the Django Models we just need to define the type of variables and their disposition in the form (in ModelAdmin) to generate a HTML form for data edition. For the GUIs that I wanted to build in my python scripts, I would like to have the same simplicity, so I could focus on the algorithms and not on GUIs developing.



**Fig 7.4.7: PyForms**

Pyforms\_GUI is a software layer that forms part of the Pyforms main library, which also includes PyForms-Web, and PyForms-Terminal. Pyforms is the Python implementation of Windows Forms, which lets you develop interactive interfaces for Windows GUI mode, Web mode, and Terminal mode.

#### **Advantages:**

- Open source
- Cross platform
- It has a minimal API, so interfaces can be easily defined with just a few lines of Python code
- You can also code advanced functionalities with minimal effort
- The code is organized in modules, ready to be reused by other applications
- Simplifies applications maintenance
- Fast and easy prototyping
- Low learning curve
- Easy install with pip

#### **7.4.8 PySides:**

PySide is the Python Qt bindings project, providing access the complete Qt 4.8 framework as well as to generator tools for rapidly generating bindings for any C++ libraries.

The PySide project is developed in the open, with all facilities you'd expect from any modern OSS project such as all code in a git repository, an open Bugzilla for reporting bugs, and an open design process. We welcome any contribution without requiring a transfer of

copyrightPySide, also known as Qt for Python, is a Python library for creating GUI applications using the Qt toolkit. PySide is the official binding for Qt on Python and is now developed by The Qt Company itself. This complete PySide2 tutorial takes you from first concepts to building fully-functional GUI applications in Python. It requires some basic Python knowledge, but no previous familiarity with GUI concepts

PySide2 is Qt for Python, offering the official Python bindings for Qt. This enables the use of Qt APIs in Python apps, and also a binding generator tool (Shiboken2) to expose C++ projects in Python.

Qt is considered the standard for GUI design against which all other Python GUI frameworks are measured. As such, PySide2 gives Python developers access to a proven collection of tools and libraries for the fast and flexible creation of user interfaces.

#### Advantages:

- Easy install via pip
- Supports both Python 3 and Python 2.7
- Cross platform
- Extensive community support and documentation
- Used by well-known companies like TomTom and Mercedes

## 7.5 Pickle module

Pickle allows for flexibility when deserializing objects. You can easily save different variables into a Pickle file and load them back in a different Python session, recovering your data exactly the way it was without having to edit your code. Pickling is used to store python objects. This means things like lists, dictionaries, class objects, and more. Generally, you will find pickling to be most useful with data analysis, where you are performing routine tasks on the data, such as pre-processing. Also, it makes a lot of sense when you're working with Python-specific data types, such as dictionaries. If you have a large dataset, for example, and you're loading that massive data set into memory every time you run the program, it may make a lot of sense to just pickle it, and then load that instead, because it will be far faster, again by 50 - 100x, sometimes far more depending on the size. Through saving the serialized object, it's nature is included, so we don't have to worry about loading things "as" strings, dictionaries, lists, etc.

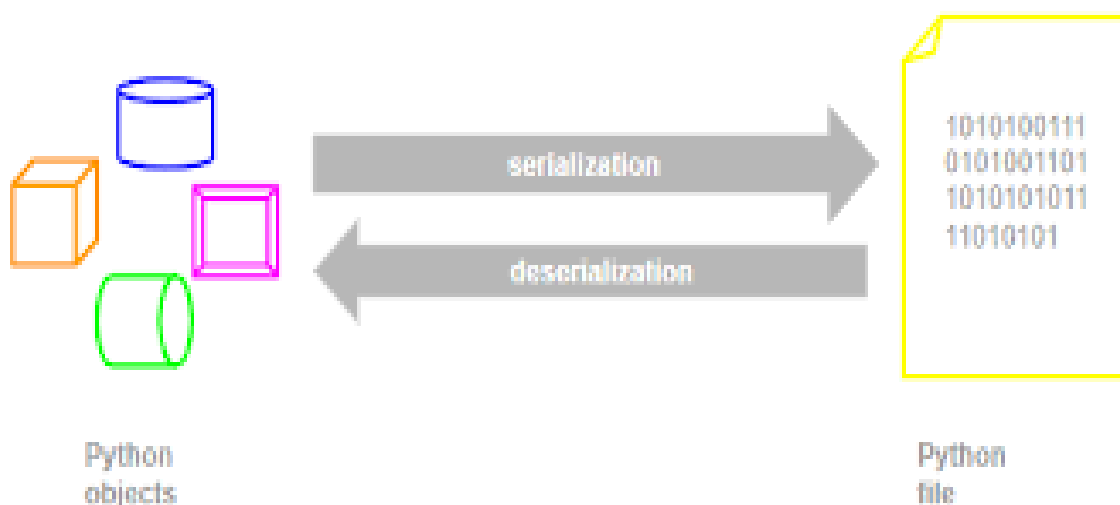


Fig 7.5: Pickle

The pickle module keeps track of the objects it has already serialized, so that later references to the same object won't be serialized again. marshal doesn't do this. This has implications both for recursive objects and object sharing. Recursive objects are objects that contain references to themselves. These are not handled by marshal, and in fact, attempting to marshal recursive objects will crash your Python interpreter. Object sharing happens when there are multiple references to the same object in different places in the object hierarchy being serialized. pickle stores such objects only once, and ensures that all other references point to the master copy. Shared objects remain shared, which can be very important for mutable objects.

## **8. SYSTEM TEST**

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

### **8.1 TYPES OF TESTS**

#### **8.1.1 Unit test**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application. It is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

#### **8.1.2 Integration test**

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfactory, as shown by successful unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

### **8.1.3 Functional test**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

- Valid Input : identified classes of valid input must be accepted.
- Invalid Input : identified classes of invalid input must be rejected.
- Functions : identified functions must be exercised.
- Output : identified classes of application outputs must be exercised.
- Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

### **8.1.4 System Test**

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

### **8.1.5 White Box Testing**

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

### **8.1.6 Black Box Test**

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box. you cannot see into it. The test provides inputs and responds to outputs without considering how the software works.

### **8.1.7 Unit Test**

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

### **8.1.8 Test strategy and approach**

Field testing will be performed manually and functional tests will be written in detail.

#### **Test objectives**

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

#### **Features to be tested**

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

### **8.1.9 Integration Test**

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g., components in a software system or – one step up – software applications at the company level – interact without error.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

#### **8.1.10 Acceptance Test**

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

## 9. PROJECT OUTCOME

### 9.1 PROJECT OUTCOME

To run code double click on 'run.bat' file to get below screen

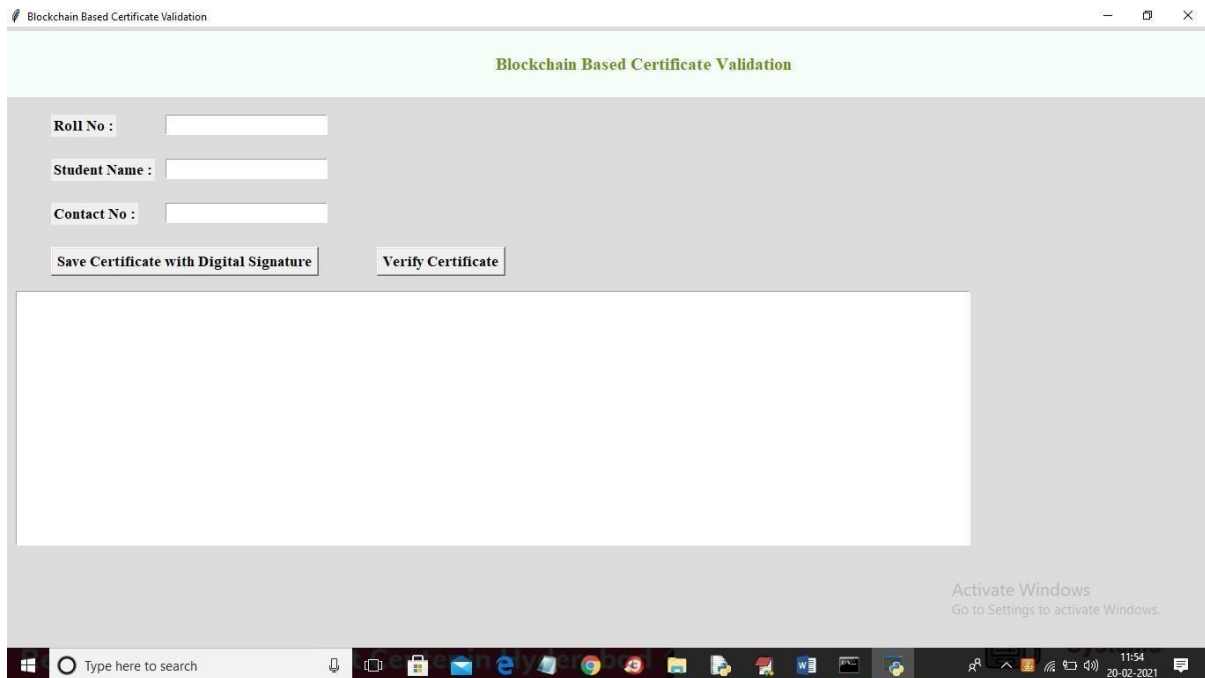


FIG 9.1.1: RUN.BAT

In above screen enter student details and then click on 'Save Certificate with Digital Signature' button to convert certificate into digital signature and then saved in Blockchain

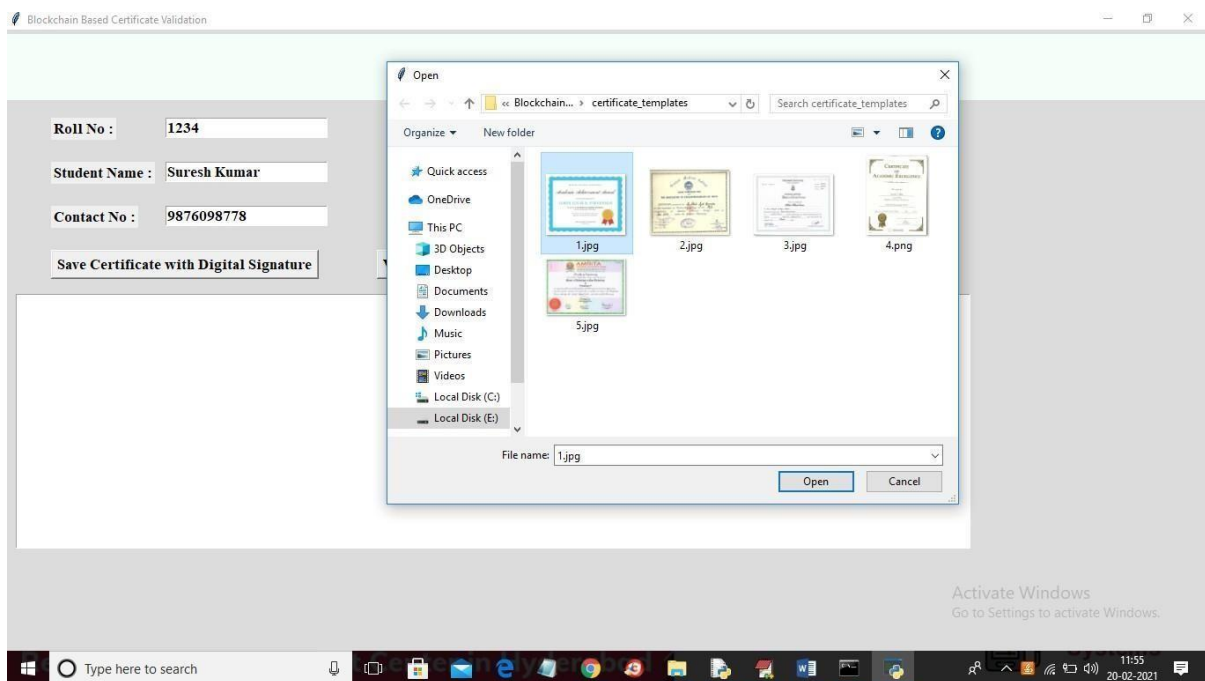
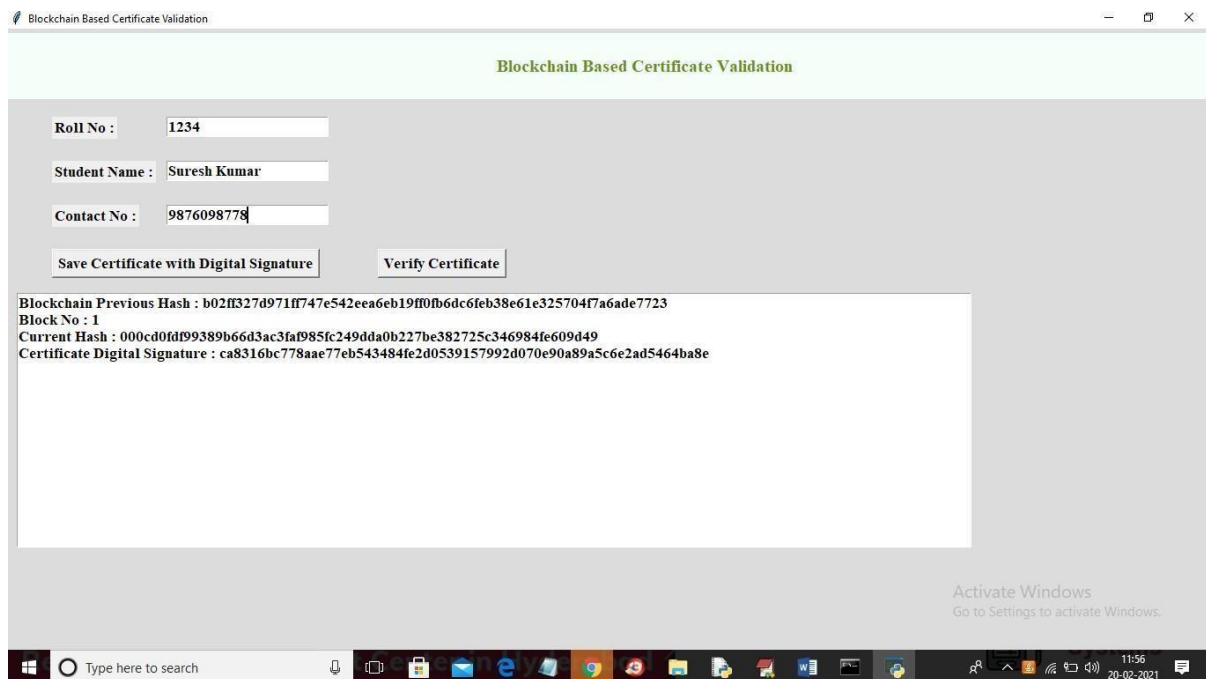


FIG 9.1.2: SAVE CERTIFICATE WITH DIGITAL SIGNATURE

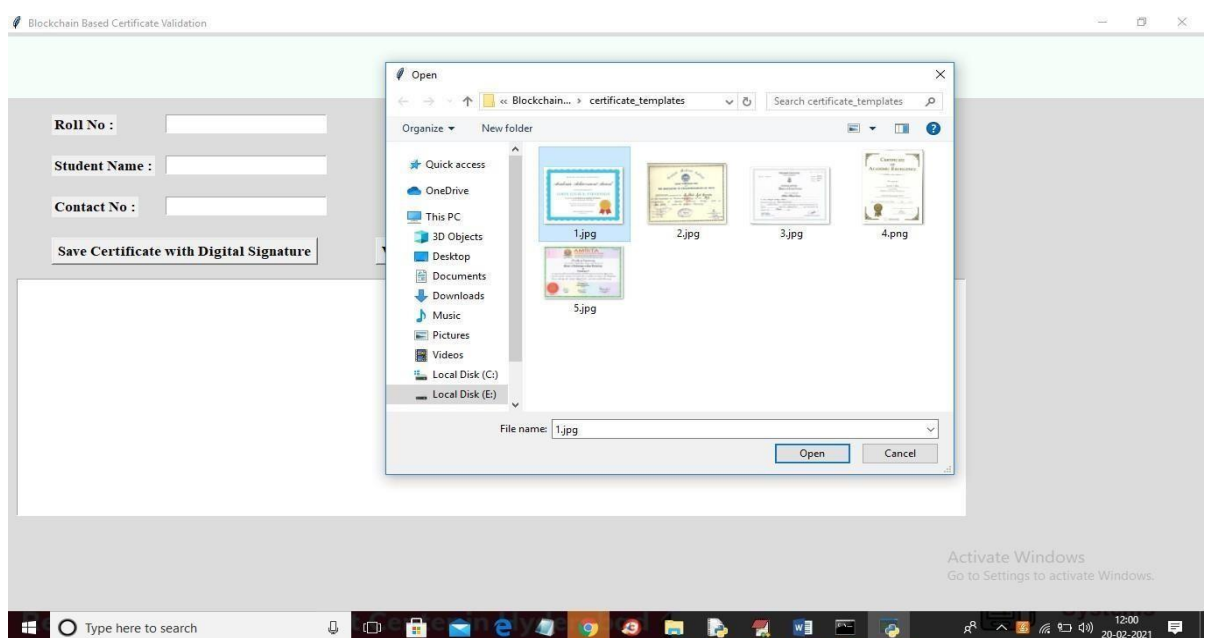


In above screen entered some student details and then click on ‘Save Certificate with Digital Signature’ button and then selecting and uploading ‘1.jpg’ file and then click on ‘Open’ button to get below screen



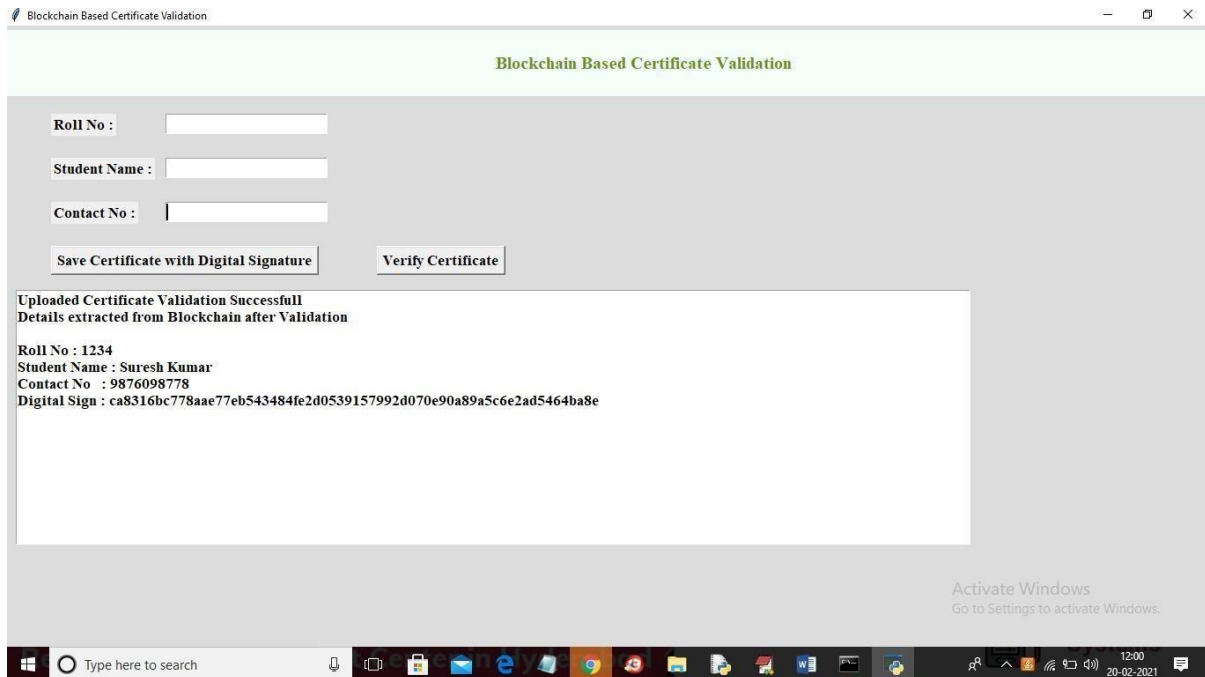
**FIG 9.1.3: VERIFY CERTIFICATE**

In above screen we can see Blockchain generated previous hash with block no 1 and its current hash and then keep on generating new blocks with each certificate upload and while running you can see that previous hash of new record will get matched with current hash of old record and this matched hash code proof that Blockchain verify old and new hash code before storing new block to confirm data is not altered. So above details stored at Blockchain and now verifier can click on ‘Verify Certificate’ button and upload same or other images to get below result.



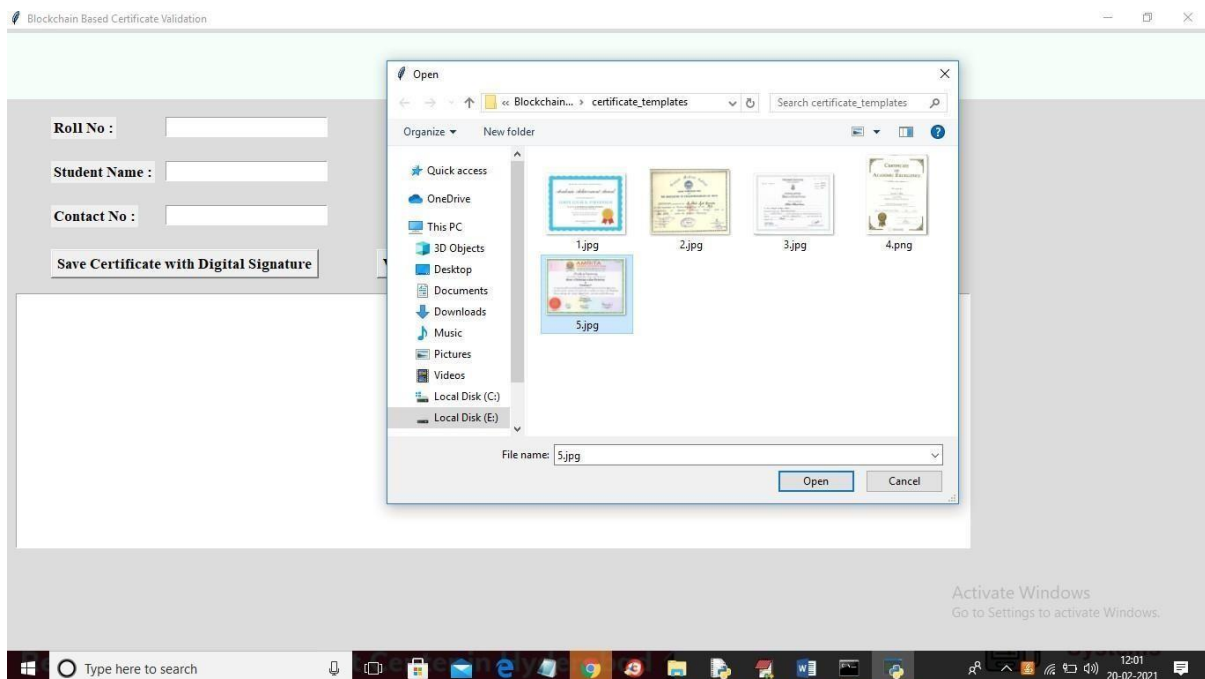
**FIG 9.1.4: VERIFY CERTIFICATE WITHOUT DETAIL**

In above screen selecting and uploading '1.jpg' file and then click on 'Open' button to get below result



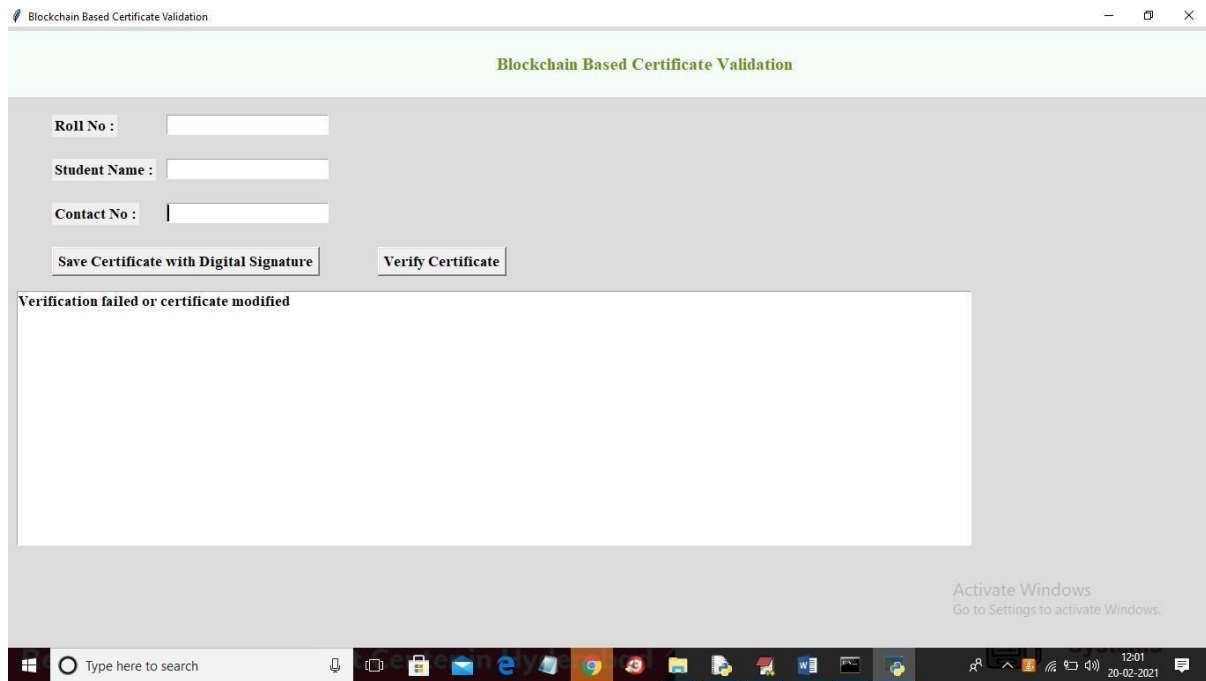
**FIG 9.1.5: VERIFY CERTIFICATE**

In above screen we uploaded same and correct image so application matched digital signature and then retrieve details from Blockchain and now try with some other image



**FIG 9.1.6: DIRECTLY VERIFYING CERTIFICATE**

In above screen selecting and uploading '5.jpg' file and then click on 'Open' button to get below result



**FIG 9.1.7: VERIFICATION FAILED**

In above screen verification got failed as uploaded certificate not matched with stored certificates in Blockchain. Similarly, you can upload any other certificate and convert them to digital signature

## 10. CONCLUSION

### 10.1 Conclusion:

In June 2016, the MIT media lab released their blockchain-based credential system which is more secure, more reliable and harder to forge, in contrast to existing technologies that based on the third-party arbitration. However, there are some serious authentication defects and vulnerable revocation mechanism which limits the prevalence and application of the project. In our project, to solve these problems and make its concept more practical, we proposed and designed a set of innovative cryptographic protocols which includes multi-signature, BTC- address-state-based revocation mechanism and trusted federated identity

Among these protocols, the multi-signature scheme most notably increases the difficulty of forging owing to the fact that each issuing progress is obliged to be signed by the majority of the academic committee members. Besides, it enhances the safety of the private keys storing for the reasons that the private keys are possessed by separated devices and people. Besides, BTC-address-based revocation mechanism improved the stability of the certificate revocation because BTC address is accessible and stable at any time. Moreover, this approach reduced the failure probability of revocation, because the cancellation process adheres the same the multi-signature algorithm, alike, involving several people. Trusted federated identity innovatively proved the authenticity of the certificate through the trusted path and federated identity. What's more, the protocol of our project can be used in other related realms such as digital right protecting and contract proof. Case in point, our protocol enables the two companies to attach their contract onto the block chain with multisignature, which is different from the traditional third party-based work mode and dispel the worries of forging credentials. Moreover, we implemented a blockchain-based certificate system, which embraced all the above protocols, by utilizing Java and JavaScript. This system has remedied the defect in Blockcerts to a certain extent, which makes the theory of blockchain-based certificate more practicable. Eventually, we conducted a series of security assessment from the perspective of operational safety, data security, network security and protocol security. The assessment outcomes provide compelling evidence that system is secured enough to meet the enterprise application standards.

Lastly, there are some limitations remained to be discussed, albeit, these considerations fall outside the scope of this paper: Our project is based on the Bitcoin blockchain, the maintenance of which relies on thousands of participants in the cryptocurrency ecosystem. Admittedly, it is imprudent to assume that the Bitcoin would work well continuously in the future because myriad types of stakeholders influence blockchain ecosystem or business model. In the years to come, we will adopt multiple blockchain sources such as Hyperledger and Ethereum to eliminate the factors of instability.

## 10.2 Future Scope:

Verification certificates using blockchain is the basic prototype for the verification of certificates. At first, we have to upload a certificate in the blockchain and it assign a digital signature and hash code to certificate and store it in the blockchain. To advance this project we can use the ML & CNN. By using that software, we just have to upload the certificate in the blockchain. After that the remaining process verifying the certificate and adding digital signature, hash code and saving the certificate by verifying the certificate clear using the CNN it helps to detail verification of certificate and secure every certificate stored in the blockchain.

Cloud Storage and the Future of Blockchain Technology. Data loss, hacking, and human mistake are all serious risks associated with centralized systems. Blockchain technology can be used to improve cloud storage security and hacker resistance, similar to how it is used in cybersecurity. Cybersecurity and the Future of Blockchain Technology. For apparent reasons, the future of blockchain technology is mostly in the area of cybersecurity. The data remains secure and verifiable despite the open and distributed nature of the Blockchain ledger. Cryptography is used to encrypt data in order to remove vulnerabilities like illegal data tampering. Governments' Use of National Digital Currencies. The year 2017 witnessed a massive increase in the value of Bitcoin which is relatively higher in comparison to other services and forms of money. Cryptocurrency has come to attain a significant position in the market as among the most valuable assets. Even with the fixed cap of 21 million units, the demand for Bitcoin will once again increase. Governments across the globe are likely to develop their own digital currencies and take part in an open market as a result. The future scope of blockchain technology may be well reflected in numerous nations' adoption of digital currency.

## 11. REFERENCES

### 11.1 References

- [1] Tengyu Yu, Blockchain operation principle analysis: 5 key technologies, iThome, <https://www.ithome.com.tw/news/105374>
- [2] JingyuanGao, The rise of virtual currencies! Bitcoin takes the lead, and the other 4 kinds can't be missed. Digital Age, <https://www.bnext.com.tw/article/47456/bitcoinether-li-tecoin-ripple-differences-between-cryptocurrencies>
- [3] Smart contractswhitepaper, <https://github.com/OSELab/learning-blockchain/blob/master/ethereum/smart-contracts.md>
- [4] Gong Chen, Development and Application of Smart Contracts, <https://www.fisc.com.tw/Upload/b0499306-1905-4531-888a-2bc4c1ddb391/TC/9005.pdf>
- [5] Weiwei He, Exempted from cumbersome auditing and issuance procedures, several national junior diplomas will debut next year.iThome, <https://www.ithome.com.tw/news/119252>
- [6] Xiuping Lin, "Semi-centralized Blockchain Smart Contracts: Centralized Verification and Smart Computing under Chains in the EthereumBlockchain",Department of Information Engineering, National Taiwan University, Taiwan, R.O.C., 2017.
- [7] Yong Shi, "Secure storage service of electronic ballot system based on block chain algorithm", Department of Computer Science, Tsing Hua University, Taiwan, R.O.C.,2017.
- [8] ZhenzhiQiu, "Digital certificate for a painting based on blockchain technology", Department of Information and Finance Management, National Taipei University of Technology, Taiwan, R.O.C., 2017.
- [9] Weiwen Yang, Global blockchain development status and trends, <http://nmarlt.pixnet.net/blog/post/>
- [10] Benyuan He, "An Empirical Study of Online Shopping Using Blockchain Technology", Department of Distribution Management, Takming University of Science and Technology, Taiwan, R.O.C., 2017.
- [11] Chris Dannen, Introducing Ethereum and Solidity, <https://www.apress.com/br/book/9781484225349>
- [12] Jan Xie, Serpent GitHub, <https://github.com/ethereum/wiki/wiki/>

## VERIFICATION OF CERTIFICATES USING BLOCKCHAIN

K. Srinivas Rao, Associate Professor, Department of Electronics and Communication Engineering,  
Kallam Harinadha Reddy Institute of Technology, Guntur, India

M. LSV Subrahmanyam, R. RVV Sai, J. Pavan Kumar, K. Balaji, K. Vamsi Student, Department of Electronics and  
Communication Engineering, Kallam Harinadha Reddy Institute of Technology, Guntur, India

### ABSTRACT:

*In this project to secure academic certificate and for accurate management and to avoid forge certificate we are converting all certificates into digital signatures and this digital signature will be stored in Blockchain server as this Blockchain server support tamper proof data storage and nobody can hack or alter its data and if by a chance if its data alter then verification get failed at next block storage and user may get intimation about data alter. In Blockchain each data will be stored by verifying old hash codes and if old hash codes remain unchanged then data will be considering as original and unchanged and then new transaction data will be appended to Blockchain as new block. For each new data storage all blocks hash code will be verified.*

**Key words:** Block chain, Hyperledger, digital certificate, hashing.

### 1. INTRODUCTION

According to, educational certificates are extremely reputable as they function as associate indicators of the human capital of their bearers. Human capital refers to the abilities, competencies, information, and aptitudes achieved through education. educational qualifications are significantly vital working things as they function as a guarantee of not simply the information, experience, and skills of the holders however additionally of their talents, responsibility, and dedication. From the attitude of the bearers, found a correlation between academic attainment levels and higher employment prospects, and economic security. recognized that educational qualifications are deemed to be real after they are presented by a university genuine when they are approved to award such certificates. Because they are thus valuable, people often lie about their academic qualifications by producing fake certificates. mentioned that within us there are presently a pair of million faux degree certificates in circulation and three hundred unauthorized universities operational. indicated that the United States has a very large number of fake institution establishments within the world followed by the UK that has concerning 270 fake institutes. Healy (2015) found that up to thirty-fifth of candidates in Australia falsified their educational credentials for the sake of employment. discovered that

almost all candidates lie a minimum of concerning some a part of their academic credentials and skill. mentioned that educational certificate fraud prices employers concerning \$ 600 billion per annum. To overcome this, we tend to mistreatment we are using technology known as Blockchain. Blockchain is that the backbone Technology of the Digital Crypto Currency Bitcoin. The blockchain could be a distributed info of records of all transactions or digital events that are dead and shared among taking part parties. every group action is verified by the bulk of participants of the system. It contains every single record of every group action. Bitcoin is that the hottest cryptocurrency associate example of the blockchain. Blockchain Technology initial came to light-weight once someone or a cluster of people name 'Satoshi Nakamoto' revealed a report on "Bitcoin: A peer to look electronic money system" in 2008. Blockchain Technology Records group action in Digital Ledger that is distributed over the Network so creating it incorrupt. something important like Land Assets, Cars, etc. is recorded on Blockchain as a group activity.

### 1.1 Building trust with Blockchain:

Blockchain enhances trust across a business network. It's not that you just can't trust those with who you conduct business with it's that you just don't got to once in operation on a Blockchain network. Blockchain builds trust through the subsequent for the attributes for this purpose.

- **Distributed:** The distributed ledger is shared and updated with each incoming group action among the nodes connected to the Blockchain. All this can be worn out real time as there's no central server dominant the information.
- **Secure:** there's no unauthorized access to Blockchain created attainable through Permissions and Cryptography.
- **Transparent:** as a result of each node or participant in Blockchain contains a copy of the Blockchain information, they need access to all or any group action information. They themselves will verify the identities while not the requirement for mediators.
- **Consensus-based:** All relevant network participants should agree that a group action is valid. this can be achieved through the utilization of agreement algorithms.
- **Flexible:** good Contracts that are dead supported bound conditions may be written into the platform. Blockchain



networks will evolve in pace with business processes and also for the future

### **1.2 Benefits of Blockchain Technology:**

- **Time-saving:** No central Authority verification is required for settlements creating the method quicker and cheaper.
- **Cost-saving:** A Blockchain network reduces expenses in many ways in which. No would like for third-party verification. Participants will share assets directly. Intermediaries are reduced. group action efforts are reduced as each participant contains a copy of the shared ledger.

### **1.3 Problem Statement:**

Finding the right candidate for the job when hiring has been a difficult work for employers or HRs of a company. One of the significant issues is the fake credentials given by the candidate like certificates. According to the survey of HireRight, about 56% of the educational credentials provided for background verification had discrepancies. Due to this, an eligible candidate may lose his opportunity to an ineligible candidate.

### **1.4 Objective:**

To define a general schema and use cryptographic methods to create Digital Academic Certificates or to build a secure, scalable blockchain-based storage system for storage of the digital certificates. or to develop an interface for data custodian and data consumer to store, manage, and deploy data certificates and upload, Authenticate certificates respectively.

### **1.5 Description:**

The physical certificates are converted into digital records. These digital records are stored in a blockchain network. Where the records once stored are immutable and cannot be changed forever due to the smart contracts. These digital records are created by the institution (issuer) and are deployed on the blockchain. A Certificate Id is sent to the user that is student or holder of the certificate. This Id can be sent to anyone intended to check or authenticate the digital certificate. For example, A company that is hiring the candidate and needs to verify the certificate authenticity of the candidate can verify it through our web application which is developed using MongoDB, React.JS.

## **2. LITERATURE SURVEY:**

### **[1] Zibin Zheng et al.**

Blockchain as a Notarization Service for information Sharing with Personal data Store: Provides a blockchain-based design to produce the credibility verification of the shared documents in period whereas maintaining necessary privacy. Discusses usage of blockchain to realize Associate in Nursing audit path of

the accesses to the shared data. Whereas keeping the audit path non-public to the people involved.

### **[2] Richard Nuetey et al.**

Block-certs and therefore the Digital Certificates project was undertaken at Media Labs Massachusetts Institute of Technology: An incubation project by the Media research laboratory Learning Initiative. Associate in Nursing the Learning Machine that builds a scheme for making, sharing, and validating bitcoin blockchain-based academic certificates. Digital certificates square measure registered on the Bitcoin blockchain, cryptographically signed, and tamper proof.

### **[3] Omar S Saleh et al.**

Authentication of User Details: The blockchain should certify users. during this case, the users square measure students, universities, institutes, employers, etc. every user in an exceedingly blockchain ledger are verified for accessing the certificate kept thereon. Authentication for users is thru a username and secret, or some system can even have multiple authentication systems like biometric, etc. as an example, the leader required to verify the certificate should initial be part of the blockchain and therefore the recipient can authorize the leader to look at the certificate and verify it.

### **[4] A Gayatri et al.**

Cheng university 'Blockchain and good Contract for Digital Certificate: Jin-chiou developed a software package so as to avoid counterfeiting certificates. thanks to the dearth of Associate in Nursing anti-forge mechanism, the graduation certificate is to be solid. so, the decentralized application was designed to support Ethereum blockchain technology. First, generate the digital certificate for the paper certificate then hash worth created for the certificate is kept within the blockchain system. Even it wants to verify the credibility of the certificate it needed another scanning app to scan the certificate. The system saves on paper, stops document forgery. however, the QRCode should be scanned with a smartphone and an online association is needed.

### **[5] A Gayatri et al.**

Project done by Ze Wang et al: Ze Wang et al designed blockchain-based certificate transparency and revocation transparency system. during this system, the certificate authority (CA) signed the certificate, and therefore the revocation standing data of the revered certificates square measure revealed by the topic (Certificate Authority). Public logs square measure want to monitor the CAs operation. this method was enforced with Firefox and nogix. this method provided the trust however Certificate validation is delayed and a false sense of security.

## **3. METHODOLOGY**

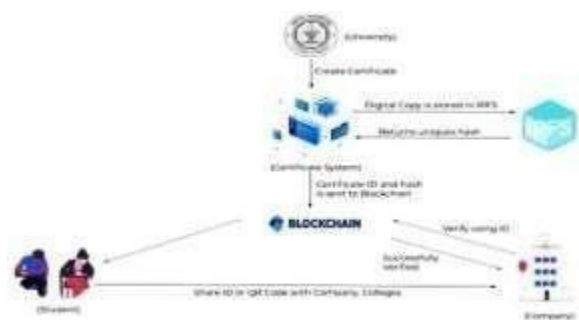
- A digital certificate is essentially a JSON Object with the necessary fields needed for our cert-issuer code to place it



on the blockchain. For which a hash can be generated and used for verification purposes.

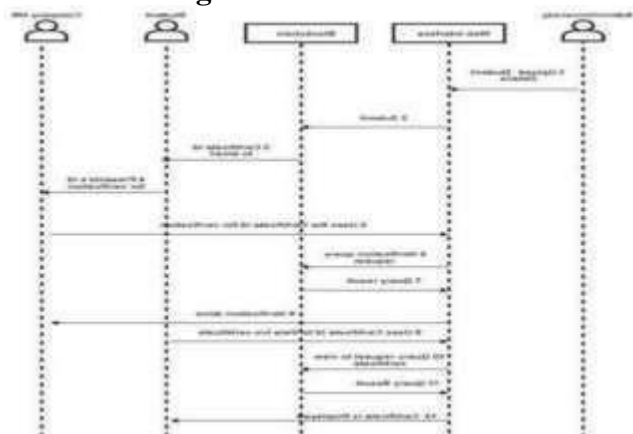
- Blockchain storage methodology Ethereum is utilized to develop an architecture to store and manage digital certificates.
- Modern web development technologies such as React-js, Node-js are utilized to build an interface that facilitates the user to view, manage, and verify documents online.

### 3.1 ARCHITECTURE DIAGRAM



A model may be a complete, basic, and simplified description, of software system design that consists of multiple views from a selected perspective or view point as shown in above figure. A read maybe an illusion of a complete system from the attitude of a connected issues. It's accustomed to describe the system from the point of view of various stakeholders like end-users, developers, project managers, and testers. Here Blockchain is the common medium of technology going to be used in the security of the data. So, we can solve the problem of counterfeit certificates.

### 3.2 Workflow Diagram:



Step 1: University will fill student academic details in a form provided using Dapp interface.

Step 2: University submits the pre-viewed form details to Blockchain

Step 3: When the certificate is uploaded into Blockchain. Certificate hash code is generated and sent to respective student.

Step 4: Students can share the Certificate hash code for certificate verification.

Step 5: With the help of hash code Company HR can verify the certificate.

Step 6: Verification of Certificate hash code query request will take place.

Step 7: Query result is displayed i.e., verified or failed.

Step 8: HR can confirm the certificate.

Step 9: The student will use his Certificate hash code to view the certificate.

Step 10: Query request to search Certificate hash code in the blockchain.

Step 11: Query result for the Certificate hash code will be displayed.

Step 12: Students can view their Certificate (Verification of certificate is optional for student).

### OUTPUT SCREEN:



### 4. APPLICATIONS

- Certificates are securely stored on Blockchain Network.
- Right person will get the Right Job Opportunity.
- Issues related to fake certificates will be avoided.
- Every time student needs to attest his/her certificate for the verification. This can be avoided.
- Certificates are easily accessible.
- Once the certificates are deployed on the Blockchain Network that will become Immutable. So, Certificate Data will not be altered.
- Manual Verification of the Certificate takes more time. This issue will be solved using Digital Certificate Verification System That will

take very less time to verify Certificate.

## **5.ADVANTAGES**

1. In Blockchain packages, Learning short time i.e., High execution speed,
2. Comparison of certificates are fast i.e., High flexible.
3. It has low memory requirements
4. Secure Server
5. As we have seen earlier, Python can be extended to other languages. You can write some of your code in languages like C++ or C. This comes in handy, especially in projects.
6. Since Python forms the basis of new platforms like Raspberry Pi, it finds the future bright for the Internet of Things. This is a way to connect the language with the real world.

## **6.CONCLUSION**

The proposed system works well on the involvement of multiple companies and institutions for the uploading of the certification details. This system plays a safer role in guarding the data secure with the help of the discussed technology. Overcomes the issues discussed above. The conversion of data into hash values play a major role in security and the interconnection of those hash values through blockchain makes it even stronger in security. The validator can be of any party, either an educational institution or a company sector. Any of the interested sectors could possibly check the received certificates for the genuinely. The certificate uploaders would need the permission and access from the smart contract owner to upload the details. The smart owner checks the requested client for they are a proper certificate issuer. After validating the certificate issuer, a unique ID and Password helps them to upload the certification details. The updation of any existing certificates would lead to notify all the other certificate issuers in the node. Thus, it is in a highly safer hand.

## **7.REFERENCES**

- [1] Tengyu Yu, Blockchain operation principle analysis: 5 key technologies, iThome, <https://www.ithome.com.tw/news/105374>
- [2] Jingyuan Gao, The rise of virtual currencies! Bitcoin takes the lead, and the other 4 kinds can't be missed. Digital Age, <https://www.bnext.com.tw/article/47456/bitcoin-ether-li-tecoin-ripple-differences-between-cryptocurrencies>
- [3] Smart contracts whitepaper, <https://github.com/OSELab/learning-blockchain/blob/master/ethereum/smart-contracts.md>
- [4] Gong Chen, Development and Application of Smart Contracts, <https://www.fisc.com.tw/Upload/b0499306-1905-4531-888a-2bc4c1ddb391/TC/9005.pdf>
- [5] Weiwei He, Exempted from cumbersome auditing and issuance procedures, several national junior diplomas will debut next year. iThome, <https://www.ithome.com.tw/news/119252>
- [6] Xiuping Lin, "Semi-centralized Blockchain Smart Contracts: Centralized Verification and Smart Computing under Chains in the Ethereum Blockchain", Department of Information Engineering, National Taiwan University, Taiwan, R.O.C., 2017.
- [7] Yong Shi, "Secure storage service of electronic ballot system based on block chain algorithm", Department of Computer Science, Tsing Hua University, Taiwan, R.O.C., 2017.
- [8] Zhenzhi Qiu, "Digital certificate for a painting based on blockchain technology", Department of Information and Finance Management, National Taipei University of Technology, Taiwan, R.O.C., 2017.
- [9] Weiwen Yang, Global blockchain development status and trends, <http://nmarlt.pixnet.net/blog/post/>
- [10] Benyuan He, "An Empirical Study of Online Shopping Using Blockchain Technology", Department of Distribution Management, Takming University of Science and Technology, Taiwan, R.O.C., 2017.
- [11] Chris Dannen, Introducing Ethereum and Solidity, <https://www.apress.com/br/book/9781484225349>
- [12] Jan Xie, Serpent GitHub, <https://github.com/ethereum/wiki/wiki/>