**GOLD PRICE PREDICTION USING**

**MACHINE LEARNING**

**A DESIGN PROJECT REPORT**

***Submitted by***

**mohamed rafeeth p**

**mukeshkar s**

**rohan chakravarthi v**

**soorej venkiteswaran r**

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

***of***

**BACHELOR OF ENGINEERING**

***in***

**COMPUTER SCIENCE AND ENGINEERING**

**K.RAMAKRISHNAN COLLEGE OF TECHNOLOGY**

(An Autonomous Institution, affiliated to Anna University Chennai and Approved by AICTE, New Delhi)

**SAMAYAPURAM – 621 112**

**July, 2023**

**K.RAMAKRISHNAN COLLEGE OF TECHNOLOGY**

**(Autonomous)**

**SAMAYAPURAM – 621 112**

**BONAFIDE CERTIFICATE**

Certified that this project report titled **“GOLD PRICE PREDICTION USING MACHINE LEARNING”** is the bonafide work of **MOHAMED RAFEETH P (811720104062), MUKESHKAR S (811720104068), ROHAN CHAKRAVARTHI V (811720104085), SOOREJ VENKITESWARAN R (811720104098)** who carried out the project under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

|  |  |
| --- | --- |
| **SIGNATURE**  Mr. R.Rajavarman M.E.,(Ph.D.,)  **HEAD OF THE DEPARTMENT**  Department of CSE  K. Ramakrishnan College of Technology  (Autonomous)  Samayapuram – 621 112 | **SIGNATURE**  Mrs. P.Silambarasi M.Tech.,  **supervisor**  ASSISTANT PROFESSOR  Department of CSE  K.Ramakrishnan College of Technology  (Autonomous)  Samayapuram – 621 112 |
|  |  |

Submitted for the viva-voce examination held on …………….

**Internal Examiner External Examiner**

**DECLARATION**

We jointly declare that the project report on **“GOLD PRICE PREDICTION USING MACHINE LEARNING”** is the result of original work done by us and best of our knowledge, similar work has not been submitted to **“ANNA UNIVERSITY CHENNAI”** for the requirement of Degree of **BACHELOR OF ENGINEERING**. This project report is submitted on the partial fulfilment of the requirement of the award of Degree of **BACHELOR OF ENGINEERING**.

|  |
| --- |
| **Signature** |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  MOHAMED RAFEETH P |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  MUKESHKAR S  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  ROHAN CHAKRAVARTHI V  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  SOOREJ VENKITESWARAN R |
|  |

Place: Samayapuram

Date:

**ACKNOWLEDGEMENT**

It is with great pride that we express our gratitude and in-debt to our institution “**K.Ramakrishnan College of Technology (Autonomous)**”, for providing us with the opportunity to do this project.

We are glad to credit honourable chairman **Dr. K. RAMAKRISHNAN**, **B.E.,** for having provided for the facilities during the course of our study in college.

We would like to express our sincere thanks to our beloved Executive Director **Dr. S. KUPPUSAMY, MBA, Ph.D.,** for forwarding to our project and offering adequate duration in completing our project.

We would like to thank **Dr. N. VASUDEVAN, M.E., Ph.D.,** Principal, who gave opportunity to frame the project with the full satisfaction.

We whole heartily thanks to **Mr. R. RAJAVARMAN**, **M.E.,(Ph.D).,** Head of the department, **COMPUTER SCIENCE AND ENGINEERING** for providing his encourage pursuing this project.

I express my deep and sincere gratitude to my project guide **Mrs. P. SILAMBARASI**, **M.Tech.,** Department of **COMPUTER SCIENCE AND ENGINEERING,** for his incalculable suggestions, creativity, assistance andpatience which motivated me to carry out this project.

I render my sincere thanks to Course Coordinator and other staff members for providing valuable information during the course.

I wish to express my special thanks to the officials and Lab Technicians of our departments who rendered their help during the period of the work progress.

**ABSTRACT**

Gold is one of the precious metals. It has been used as currency, for jewellery and other purposes. It is used as medium for money or exchange because of its limited supply and high value. This metal’s scarcity and difficulty in extraction made it a valuable commodity. It also reflects the country’s economical strength and hence many companies and individuals started to invest in gold reserves. Due to its increasing value, many people considered gold as an attractive investment. Gold is preferred as protective asset by investors because of their negative expectations regarding the current situation in the foreign exchange and capital markets. Investors also consider gold as an asset to rely on, when the desirable profits are not achieved by the world capital markets. Since gold is stored and accumulated over years, the influence of an year’s production on its price is less. The raise of gold prices and fall of prices in other markets has attracted more investors to invest in gold market. These changes in the price of gold made the investments risky and a fear has been developed that these prices would decrease. There are several number of studies analyzing the relation between the gold price andother economic variables. Understanding the relation between these variables helps the investors to take better decisions. Hence, we use machine learning algorithms such as multiple linear regression, random forest and gradient boosting for analyzing the relation between the variables and predict the gold price.

**TABLE OF CONTENTS**

**CHAPTER TITLE PAGE NO.**

**ABSTRACT v**

**LIST OF TABLES vi**

**LIST OF FIGURES ix**

**LIST OF ABBREVIATIONS x**

1. **INTRODUCTION 1**

1.1 OVERVIEW 1

1.2 PROBLEM STATEMENT 2

1.3 OBJECTIVES 2

1.4 IMPLICATION 2

1. **LITERATURE SURVEY 4**

2.1 BLOCKCHAIN-BASED ACADEMIC CERTIFICATE VERIFICATON SYSTEM 4

2.2BLOCKCHAIN-BASED SYSTEM FOR DOCUMENT VERFICATION AND ANTHENTICATION 4

2.3 SERCURING DOCUMENTS USING BLOCKCHAIN TECHNOLOGY 5

2.4 BLOCKCHAIN FOR ACADEMIC CREDENTIALS SECURE,DECENTRALIZED,TRANSPARENT 5

1. **SYSTEM ANALYSIS 6**

3.1 EXISTING SYSTEM 6

3.2 PROPOSED SYSTEM 6

3.3 APPROACH USED 6

1. **THEORETICAL CONSIDERATION 7**

4.1 HISTORICAL INTRODUCTION 7

4.2 BLOCKCHAIN TECHNOLOGY 8

4.2.1 OVERVIEW OF BLOCKCHAIN ARCHITECTURE 9

4.2.2 CONSENSUS PROTOCOL 11

4.2.3 BLOCKCHAIN TYPES 12

4.2.4 BLOCKCHAIN FEATURES 13

4.2.5 TRANSACTIONS 13

4.2.5.1 GAS 14

4.2.5.2 DIGITAL SIGNATURE 14

4.2.5.2.1 COMPONENTS OF A

DIGITAL SIGNATURE 15

4.2.5.2.2 PUBLIC KEY INFRASTRUCTURES 15

4.2.5.2.2.1 PUBLIC KEY 16

4.2.5.2.2.2 PRIVATE KEY 16

4.2.5.2.2.3 HASHING 16

4.2.6 BLOCKCHAIN PLATFORMS 18

4.3 ETHEREUM BLOCKCHAIN 20

4.3.1 SMART CONTRACT 20

4.3.1.1 EVM 20

4.3.1.2 SOLIDITY 21

4.3.1.2.1 ABI 21

4.3.1.3 REMIX 22

4.3.1.4 GANACHE 22

4.3.1.5 TRUFFLE 22

4.3.2 ETHEREUM WALLET 23

4.3.2.1 METAMASK WALLET 23

4.3.3 WEB3.JS 23

4.4 POLYGON NETWORK 23

4.5 IPFS 24

4.6 PROOF OF EXISTENCE 24

1. **MODULE DESCRIPTION 25**

5.1 UI DESIGN 25

5.2 ADMIN PANEL 25

5.3 UPLOAD PAGE 25

5.4 VERFICIATION 26

5.5 DELETION 26

1. **SYSTEM SPECIFICATION 27**

6.1 HARDWARE REQUIREMENTS 27

6.2 SOFTWARE REQUIREMENTS 27

1. **METHODOLOGY 28**

7.1 INTRODUCTION 28

7.2 SYSTEM ARCHITECTURE 28

7.3 IMPLEMENTATION DETAILS 30

7.3.1 SMART CONTRACT 30

1. **CONCLUSION AND FUTURE ENHANCEMENT 42**

8.1 CONCLUSION 42

8.2 FUTURE ENHANCEMENT 42

**APPENDICES I**

**APPENDICES II**

**REFERENCES**

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **FIGURE NO.** | **TITLE** | **PAGE NO.** |
| 4.1  4.2  4.3  4.4  4.5  4.6  7.1  7.2  7.3  7.4  7.5  7.6  7.7 | LEGACY & DISTRIBUTED LEDGER  HOW DOES BLOCKCHAIN WORKS  SIMPLIED STRUCTURE OF BLOCKCHAIN  ANATOMY OF DIGITALLY SIGNED DOCS  CRYPTOGRAPHIC HASH FUNCTION  RECORDING AND VERIFYING DOCUMENTS  WORKFLOW OF THE IMPLEMENTATION  ARCHITECTURE & MECHANISM OF SYSTEM  FUNCTION OF ADDING A DOCS EXPORTER  FUNCTION OF DELETING A DOCS EXPORTER  FUNCTION OF EXPORT A DOCUMENT  FUNCTION OF VERIFY A DOCUMENT  FUNCTION OF DELETING A DOCUMENT | 8  10  12  16  17  24  28  29  34  36  37  39  41 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**LIST OF ABBREVIATIONS**

**ABBREVIATIONS**

|  |  |  |
| --- | --- | --- |
| **DLT** | - | Distributed Ledger Technology |
| **PoW** | - | Proof of Work |
| **PoS** | - | Proof of Stake |
| **SHA** | - | Secure Hash Algorithm |
| **EVM** | - | Ethereum Virtual Machine |
| **ABI** | - | Application Binary Interface |
| **IDE** | - | Integrated Development Environment |
| **NPM** | - | Node Package Manager |
| **P2P** | - | Peer-to-Peer |
| **QR** | - | Quick Response |

**CHAPTER 1**

**INTRODUCTION**

* 1. **PROJECT BACKGROUND**

Historically, gold had been used as a form of currency in various parts of the world including USA [5]. In recent times also, gold has maintained its value and has been used as a means for assessing the financial strength of a country.

Big investors have also been attracted to this precious metal and invested huge amounts in it. Recently, emerging world economies, such as China, Russia, and India have been big buyers of gold. Whereas USA, South Africa, and Australia are among the big seller of this commodity [8]. Chinese and Indian traditional events also affect the price of the gold. In that time more money is poured for purchase of this commodity. Small investors also find this commodity for safe investment rather than alternate investment options, which bear in-built investment risks. Internal financial conditions of the aforementioned countries play a vital role for setting spot rates for gold.

Governmental investments in gold are largely decided by their financial conditions, and interest rates, as they are indicators of the strength of their economy. When US interest rates become lower, more economic activity is witnessed in US, thus capital inflows in gold market are observed. Similarly, when interest rates lowered in China from (2010) to (2016), it bought gold aggressively.

* 1. **GOLD PRICE PREDICTION**

There are so many studies dealing with the price of gold in the world. Although various different kind of variables are used in these studies, it is predict the gold price. Gold price prediction by relationship between gold price and selected factors influencing it , namely date, stock value, current gold price ,united state oil price, current silver price, currency. Medium (EUR/USD) using by random forest regression algorithm..

* 1. **FUNDAMENTAL ANALYSIS**

Money supply is one of the important factor to consider for fundamental analysis of gold markets. Money supply is the entire stock of currency and other liquid instruments circulating in a country’s economy as of a particular time.Money supply includes safe assets, such as cash, coins, and balances held in checking and savings accounts that businesses and individuals can use to make payments or hold as short-term investments.An increase in the supply of money typically lowers interest rates, which in turns generates more investment and puts more money in the hands of consumers, thereby stimulating spending. Businesses respond by ordering more raw materials and increasing production.The increased business activity raises the demand for labour.The opposite Can occur if the money supply falls or when its growth rate decline.

* 1. **TECHNICAL ANALYSIS**

Gold has been considered a highly valuable commodity for millennia and the gold price is widely followed in financial markets around the world. Mostly quoted in US Dollars (XAU/USD), gold price tends to increase as stocks and bonds decline. The metal holds its value well, making it a reliable safe-haven. It straded constantly based on the intra-day spot rate. Improve your technical analysis of live gold prices with the real-time XAU/USD chart, and read our latest gold news, expert analysis and gold price forecast.

* 1. **MACHINE ANALYSIS**

Machine learning is the technique of examine the data that automates analytical model to occur an make with a building. It is related to artificial intelligence based on the aim that systems can learn from data, find patterns and make conclusion with very minimum human interaction. With the advent of new computing technologies, the present day machine learning is totally different from how it was in its inception. Machine learning was made from the pattern recognition methodology and from the theory that machine learning is possible without being programmed for specific tasks. The most important aspect of machine learning is the iterative aspect as the data models are uncover to new data they adapt independently. With the advent of new computing technologies, the present day machine learning is totally different from how it was in its inception. Machine learning was made from the pattern recognition methodology and from the theory that machine learning is possible without being programmed for specific tasks. The most important aspect of machine learning is the iterative aspect as the data models are uncover to new data they adapt independently.

**CHAPTER 2**

**LITERATURE SURVEY**

**2.1 TITLE:** **AN**  **Prediction of Gold Price Based on WT-SVR and EMD-SVR Mode**

**AUTHOR: J.Risk Financial Manag. 2021,**

**YEAR & PUBLICATION: August 27, 2020, IEEE**

Gold is often used by investors as a hedge against inflation or adverse economic times. Consequently, it is important for investors to have accurate forecasts of gold prices. This paper uses several machine learning tree-based classifiers (bagging, stochastic gradient boosting, random forests) to predict the price direction of gold and silver exchange traded funds. Decision tree bagging, stochastic gradient boosting, and random forests predictions of gold and silver price direction are much more accurate than those obtained from logit models. For a 20-day forecast horizon, tree bagging, stochastic gradient boosting, and random forests produce accuracy rates of between 85% and 90% while log it models produce accuracy rates of between 55% and 60%. Stochastic gradient boosting accuracy is a few percentage points less than that of random forests for forecast horizons over 10 days.

**2.2 TITLE: Spot Gold Price Prediction Using Financial News Sentiment Analysis.**

**AUTHOR:** [**Zhou Junjie, Paolo Mengoni,**](https://www.computer.org/csdl/search/default?type=author&givenName=Zhou&surname=Junjie)

**YEAR & PUBLICATION: August 27, 2020, IEEE**

Data analytic helps investors to make prediction of the financial market. Financial news or other related information is an invaluable asset for investors' accurate prediction and efficient decision. Most researchers focus on stock price prediction. Commodity market, and especially Spot Gold, has not been investigated deeply in literature

**2.3 TITLE: Gold price prediction in times of financial and geopolitical uncertainty: A machine learning approach.**

**AUTHOR:** [**Raland Matenggo Akbar Bhakti Abu Bakar**](https://www.researchgate.net/profile/Raland-Matenggo-Abu-Bakar)

**YEAR: 2018**

Gold has always occupied a predominant place in country's economies, and among populations. Owing to its characteristics, it is used as a hedging tool or can act as a safe haven in turmoil conditions. The aim of this study is to explore the relationship of gold price with various explanatory variables that tend to be considered as indicators of financial and geopolitical crises.

**2.4 TITLE: Regression and Hidden Markov Models for Gold Price Prediction**

**AUTHOR: Li Shen,Kun Shen,Yixin Chen,**

**YEAR & PUBLICATION: July 13, 2021, IEEE**

In the long run, gold price is positively related to inflation rates because gold is perfect asset to hedge against inflation. In the short run, gold price fluctuates a lot. Many factors can cause gold price volatility, such as economic and political uncertainties, exchange rates, interest rates and so on. Here we try several models to predict monthly gold prices, including linear regression model and ARIMA model. We also try to predict monthly gold returns with hidden Markov model.

**2.5 TITLE: Prediction of Gold Price Based on WT-SVR and EMD-SVR Mode**

**AUTHOR:** [**Yang Jian-Hui Dou Wei**](https://www.computer.org/csdl/search/default?type=author&givenName=Yang&surname=Jian-Hui)

**YEAR & PUBLICATION: February 9, 2012, IEEE**

The current gold market show a high degree of nonlinearity and uncertainty, in order to predicted the gold price, Empirical Mode Decomposition (EMD) is introduced, the use the EMD orthogonal decompose the special functions into a finite number of independent intrinsic mode functions (IMFs), then Grouping the IMFs according different frequencies, using support vector regression (SVR) to predict each IMF group, at last plus each forecasting value of equal weighted will get the final prediction. Comparative analysis with the traditional practice is relatively mature wavelet transform (WT), WT decompose the function into some signal, then using SVR to predict detail signals and approximation signal, at last plus each forecasting parts will get the final prediction. Empirical studies show that: EMD has more accurate prediction than WT. This method provides a new powerful analytical tool for the gold price Prediction, an important guiding tool in theory and practice.

**2.6 TITLE: Gold and Diamond Price Prediction Using Enhanced Ensemble Learning.**

**AUTHOR: Avinash Chandra Pandey, Shubhangi Misra,Mridual Saxena,**

**YEAR & PUBLICATION: February 16, 2019, IEEE**

Precious metals like diamond and gold are in high demand due to their monetary rewards. Therefore, various techniques are generally employed to forecast prices of diamonds and precious metals with the aim of fast and accurate results. The prices fluctuate daily making it difficult to predict the next future value. Hence, by examining the pattern of previous prices we can apply regression models for future prediction.

**­CHAPTER 3**

**SYSTEM SPECIFICATION**

**3.1 H/W SYSTEM CONFIGURATION**

* Processor-pentium-IV
* RAM-4 GB(min)
* Hard disk-20GB

**3.2 S/W SYSTEM CONFIGURATION**

* Operating system:window 7 and above
* Front end:python
* Back end:CMD
* Plat form:jupyter

#### 3.3 SOFTWARE DESCRIPTION

Python is an interpreter, high level, general purpose programming language. Python is dynamically typed, and garbage collected.it support multiple programming paradigms, including procedural, object oriented, and functional programming language. python is often described a Batteries included due to its comprehensive standard library. Python was conceived in late 1980’s as a successor to the ABC language. python2.0, released in 2000, introduced features like list comprehension and garbage collection system capable of collecting reference cycles. Python 3.0, released in 2008, was a major revision of the language that is naot completely backward compatible, and much python 2 code does not run unmodified on python 3.

**LIBRARIES**

Python's large standard library, commonly cited as one of its greatest strengths, provides tools suited too many tasks. For Internet-facing applications, many standard formats and protocols such as MIME and HTTP are supported. It includes modules for creating graphical user interfaces, connecting to relational databases, generating pseudo random numbers, arithmetic with arbitrary- precision decimals, manipulating regular expressions, and unit testing.

Some parts of the standard library are covered by specifications (for example, the Web Server Gateway Interface (WSGI) implementation wagered follows PEP 333), but most modules are not. They are specified by their code, internal documentation, and test suites (if supplied). However, because most of the standard library is cross-platform Python code, only a few modules need altering or rewriting for variant implementations.

**DEVELOPMENT ENVIRONMENT**

Most python implementation (including python) include a read- evalue-print loop (REPL),permitting them to function as a command line interpreter for which the user enters statement sequentially and receivers results immediately.

Other shells, including IDLE and I Python, add further abilities such as auto-completion session state retention and syntax highlighting.As well as standard desktop integrated development environments,there are web browser-based IDE’s;sage maker math (intended for developing science and math-related python programs).A browser-based IDE and hosting environment and canopy IDE,a commercial python IDE emphasizing scientific computing.

1. **Sign up for an AWS Account:** If you don't already have an AWS account, sign up for one at https://aws.amazon.com/. This will give you access to the AWS Management Console and allow you to utilize AWS services.

2. **Choose a Programming Language**: Decide on the programming language you will use for your development. Popular choices for data analysis and machine learning include Python and R.

3. **Set up AWS Command Line Interface (CLI):** Install the AWS CLI on your local machine to interact with AWS services from the command line. You can find installation instructions at https://aws.amazon.com/cli/.

4. **Set up an AWS S3 Bucket**: Create an S3 bucket to storeyour data files, code, and any other resources you need for your gold price prediction project. You can do this through the AWS Management Console or by using the AWS CLI.

5. **Install Development Tools**: Set up your preferredintegrated development environment (IDE) for coding. For Python development, popular choices include Anaconda, Jupyter Notebook, PyCharm, or Visual Studio Code. Install the necessary libraries and packages for data analysis, machine learning, and AWS SDKs for your chosen programming language.

6. **Data Exploration and Analysis:** Use your chosen development environment and libraries to explore and analyze the historical gold price data. This may involve data cleaning, preprocessing feature engineering, and visualization to gain insights into the data.

7. **Choose and Implement Predictive Models:** Select the appropriate predictive models for gold price prediction, such as linear regression, random forest, or other machine learning algorithms. Implement the models using your chosen programming language and libraries.

8. **Utilize AWS Services:** Integrate AWS services into your development environment as needed. Use Amazon Athena for querying and analyzing data stored in your S3 bucket using SQL-like queries. Use Amazon SageMaker for training and deploying machine learning models. Use Amazon SageMaker for training and deploying machine learning models. Use Amazon CloudWatch for monitoring and logging system performance.

9. **Test and Validate**: Test your predictive models using appropriate evaluation techniques and validation methods. Ensure that the models provide accurate predictions and are performing as expected.

**CHAPTER 4**

**SYSTEM ANALYSIS**

**4.1** **EXISTING SYSTEM**

Until now by having a look at these existing papers we’ve come to know that they all have been dependent on the previously executed technical methods, which creates a major impact in the accuracy of the values.

Introduction to Gold Price Prediction: Provide a brief overview of the project's objective and the significance of predicting gold prices in financial markets.

System Architecture: Describe the overall architecture of your system, highlighting the key components and their interactions. Explain how AWS services are utilized within the architecture.

Data Collection and Preprocessing: Discuss the process of collecting historical gold price data and any other relevant data sources. Explain how the data is preprocessed and prepared for analysis, including steps such as data cleaning, normalization, and feature engineering.

AWS Services Utilized: Outline the AWS services that are employed in your system. Amazon S3 (Simple Storage Service): Explain how S3 is used for storing the historical data and making it accessible to the analysis pipeline.

AWS Glue: Describe how AWS Glue is used for data cataloging, automated schema inference, and data transformation.

Amazon Athena: Discuss how Athena is utilized for querying and analyzing the data stored in S3 using SQL-like queries.

Amazon Machine Learning Services: Explain the usage of AWS machine learning services like Amazon SageMaker for training and deploying predictive models.

Amazon Cloud Watch: Discuss how Cloud Watch is used for monitoring system performance, logging, and tracking key metrics.

Predictive Modeling: Detail the predictive modeling techniques employed, such as linear regression, random forest, or other machine learning algorithms. Explain how these models are trained, evaluated, and optimized using the gold price data.

Performance Evaluation: Discuss the metrics and methods used to evaluate the performance of your predictive models. Explain how you assess the accuracy, precision, recall, or any other relevant metrics to measure the effectiveness of your gold price predictions

Scalability and Elasticity: Highlight how your system utilizes the scalability and elasticity features provided by AWS to handle large volumes of data and perform computations efficiently.

Cost Analysis: Provide an analysis of the costs associated with running the system on AWS. Discuss the pricing models of the AWS services utilized and provide insights on optimizing costs while maintaining system performance.

Conclusion and Future Enhancements: Summarize the key findings of your system analysis and discuss potential areas for improvement. Highlight any future enhancements or additional features that could be incorporated into the system.

References: Include a list of references used in your research and system analysis.

Remember to tailor the content and depth of your report according to the specific requirements and guidelines provided by your college or university.

**ALGORITHM USED FOR GOLD PRICE PREDICTION**

**ARIMA -** ARIMA is an acronym for “autoregressive integrated moving average.” It’s a model used in statistics and econometrics events that happen over a period of time**.**

**SVM -** SVM’s are used in application like handwriting recognition, intrusion detection, face detection, email classification, and over a period of time.

**LINEAR REGRESSION -** Linear Regression is a machine learning algorithm based on supervised learning. It performs a regression task. Regression models a target prediction value based on independent variables.

**RANDOM FOREST-**Random forests or random decision forests is an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time.

**GRADIENT BOOSTING -** Gradient Boosting Algorithm is generally used when we want to decrease the Bias error. Gradient Boosting Algorithm can be used in regression as well as classification problems. In regression problems, the cost function is MSE whereas, in classification problems, the cost function is Log-Loss.

**CVM** - CVMs are algorithms which work on the basis of Minimum Enclosing Ball concept. It detects the attacks like: U2R attack, R2L attack, Probe attack and Dos attack. For each type of attack, a CVM classifier is model. KDD Cup'99 dataset is used for training and testing the classifiers.

**BEST ALGORITHM**

**LINEAR REGRESSION AND RANDOM FOREST :**

Gold Price Prediction has always been a hot topic, especially when the prevalence of Covid- 19 calls for ways involving many places where in some country is critical. However, although there are various algorithms for Gold Price Prediction,there is limited research in determining the performance of these algorithms using scientific evidence.

The experimental results demonstrate that Linear Regression and Random forest achieved superior performance with successful probability of 95.5%.

Linear Regression can be used to build a model that predicts the price of gold based on input features such as historical gold prices, economic indicators, or other relevant factors. In this case, the target variable would be the gold price, and the input features would be the factors that are expected to influence gold prices.

**4.2** **BLOCKCHAIN TECHNOLOGY**

In Blockchain is a decentralized database (distributed ledger) that records transactions or even digital events that have been executed and shared among the participating parties. Each transaction can be verified by mode of consensus of a majority of the members in the system. And the recorded information in the chain can never be altered or erased. Each of the transactions made on the blockchain is certain and verifiable. To use a basic analogy, it is easy to steal a book from a secluded room that has no people than stealing the same book from a big hall containing many people observing you.

A picture containing line, diagram, design

Description automatically generated

Figure 4.1: (a) Legacy Ledgers (Centralized); (b) Distributed Ledger.

As previously mentioned, every node in the network has an identical copy of the blockchain. Every change performed in one of them is sent to every other node. Unfortunately, a big problem arises from it. How to guarantee that the changes made are valid/trustworthy? This is famously known as the Byzantine Fault. To solve this problem, there is the use of consensus algorithms.

**4.2.1** **OVERVIEW OF BLOCKCHAIN ARCHITECTURE:**

A blockchain is a ledger linking sequential “blocks” of transactions whereby:

* Every person who wishes to trade any asset across a private or public network requires access to the network. This access occurs via a software application that mediates between user and blockchain. The software application, often called a “wallet,” can be installed directly on a device or accessed via a web browser. Depending on how it is designed, a blockchain wallet can be used to send and/or receive digital assets. Some wallets allow for direct transacting without a mediating third-party, while other wallets are run by third parties who maintain custodianship of users’ digital assets on their behalf.
* Those users wishing to participate in validating transactions through consensus must generally to install the blockchain software on their device. This is used to write to the ledger, store an entire copy of the entire ledger and keep all the copies of the ledger perfectly synchronised. Because public blockchains allow anyone to install the software and have a copy of the entire ledger, anyone can transact directly on the Blockchain within the network, and no third parties can impose conditions for access. In permissioned blockchains, a centralized authority determines who has access to run a node and participate in the consensus process.
* The transaction-records, or blocks, in a blockchain are linked together cryptographically, rendering them tamper-proof. Unlike records in digital databases, which can be altered, once a transaction is recorded and timestamped on the Blockchain, it is impossible to alter it, or delete it.
* The blockchain records the fact of the transaction, that is, what has been transferred, the parties involved, as well as structured information (metadata) related to the transaction and a cryptographic hash (“digital fingerprint”) of transaction content. This unique signature is used to verify transactions later: if someone alters the transaction content, its resulting unique code no longer matches the version that is on the chain, and the blockchain software will highlight the discrepancy.
* All parties involved in a transaction, and only those parties, must provide their consensus before a new transaction record is added to the network. All other nodes in the network will only verify that the two parties have the appropriate capacity to enter into the transaction. Thus, as soon as one party agrees to send the asset, and the other party agrees to receive the asset, and the nodes verify that each party has the capacity to conduct the transaction, it is completed.
* All computers in the network continually and mathematically verify that their copy of the blockchain is identical to all the other copies on the network. The version running on the majority of computers is assumed to be the ‘real’ version, so the only way to ‘hack’ the records would be to take control of over half of the computers on the network. For a blockchain running on thousands (or even, in the future, millions) of computers, as public blockchains like Bitcoin and Ethereum do, this would-be a near-impossible task. Destroying the ledger entirely would require deleting every copy of it in the world.

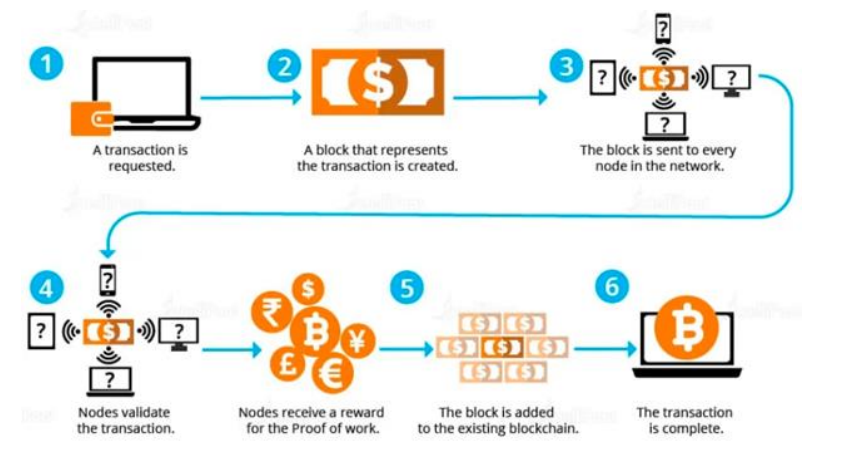


Figure 4.2: How does blockchain work.

**4.2.2** **CONSENSUS PROTOCOL:**

By consensus, we mean that a general agreement has been reached. Consider a group of people going to the cinema. If there is not a disagreement on a proposed choice of film, then a consensus is achieved. In the extreme case the group will eventually split.

In regards to blockchain, the process is formalized, and reaching consensus means that at least 51% of the nodes on the network agree on the next global state of the network.

Consensus mechanisms (also known as consensus protocols or consensus algorithms) allow distributed systems (networks of computers) to work together and stay secure.

For decades, these mechanisms have been used to establish consensus among database nodes, application servers, and other enterprise infrastructure. In recent years, new consensus mechanisms have been invented to allow crypto economic systems, such as Ethereum, to agree on the state of the network.

A consensus mechanism in a crypto economic system also helps prevent certain kinds of economic attacks. In theory, an attacker can compromise consensus by controlling 51% of the network. Consensus mechanisms are designed to make this "51% attack" unfeasible. Different mechanisms are engineered to solve this security problem in different ways.

There are different kinds of consensus mechanism algorithms, each of which works on different principles, the most famous are proof of work (PoW) and proof of stake (PoS).

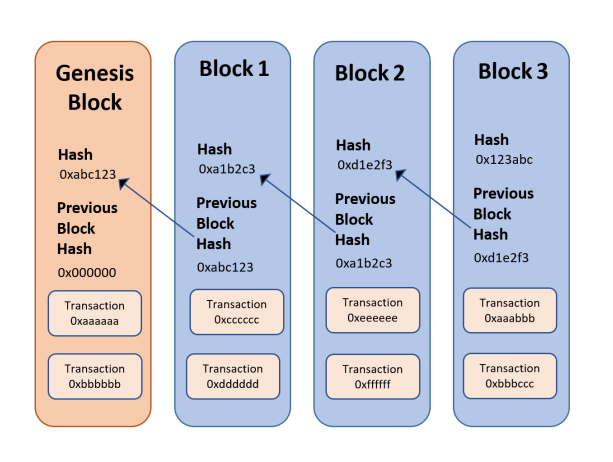


Figure 4.3: Simplified structure of Blockchain.

**4.2.3** **BLOCKCHAIN TYPES:**

Broadly speaking, there are three different types of blockchain solutions which may be applied, each of which has significant differences in architecture and governance:

1. Public blockchains:

Public blockchains are open for anyone to download, run and transact on. Solutions built using this rely on public consensus to reach decisions, and typically may run on up to millions of machines. Thus, public blockchains produce maximum immutability, decentralisation and transparency – however, this is at the cost of high inefficiency in the form of high storage costs, high electricity usage, as well as low transaction speed and volume.

1. Private blockchains:

Private blockchains are by invitation only, and operate according to a set of rules put in place by those inviting. Such a blockchain may be used by a small number of parties to trade exclusively amongst themselves, or it may be open to anyone to transact upon, but only allow a select group of users to change the rules and/or to validate transactions. Effectively, a private blockchain reduces the immutability, and transparency of the chain, and is highly centralised (while still offering these advantages more than a traditional database) – however, the reduced number of parties involved means that the chain itself tends to be much smaller and specialised – leading to high efficiency, high transaction volume and speed, and consequently lower costs and resources usage.

1. Consortium Blockchains:

Consortium blockchains are effectively a hybrid of the two models. A consortium blockchain is a private blockchain, i.e. by invitation only, but all persons invited have equitable voting rights, with decisions taken by consensus. Thus, from a governance perspective it keeps the decentralised nature of a public blockchain. In terms of immutability, transparency and resource usage it provides a midway between the features of private and public chains.

**4.2.4** **BLOCKCHAIN FEATURES:**

Blockchain technology has many features:

* Improved accuracy by removing human involvement in verification.
* Cost reductions by eliminating third-party verification.
* Decentralization makes it harder to tamper with.
* Transactions are secure, private, timestamped, and efficient.
* Immutable.
* Transparent technology.
* Provides a banking alternative and a way to secure personal information for citizens of countries with unstable or underdeveloped governments.

**4.2.5** **TRANSACTIONS:**

A blockchain is a globally shared, transactional database. This means that everyone can read entries in the database just by participating in the network. If you want to change something in the database, you have to create a so-called transaction which has to be accepted by all others. The word transaction implies that the change you want to make (assume you want to change two values at the same time) is either not done at all or completely applied. Furthermore, while your transaction is applied to the database, no other transaction can alter it . As an example, imagine a table that lists the balances of all accounts in an electronic currency. If a transfer from one account to another is requested, the transactional nature of the database ensures that if the amount is subtracted from one account, it is always added to the other account. If due to whatever reason, adding the amount to the target account is not possible, the source account is also not modified. Furthermore, a transaction is always cryptographically signed by the sender (creator). This makes it straightforward to guard access to specific modifications of the database. In the example of the electronic currency, a simple check ensures that only the person holding the keys to the account can transfer money from it.

**4.2.5.1**  **GAS:**

Gas refers to the unit that measures the amount of computational effort required to execute specific operations on the Ethereum network. Since each Ethereum transaction requires computational resources to execute, each transaction requires a fee. Gas refers to the fee required to conduct a transaction on Ethereum successfully.

**4.2.5.2 DIGITAL SIGNATURE:**

Digital Signature means binary code that, like a handwritten signature, authenticates and executes a document and identifies the signatory. A digital signature is practically impossible to forge and cannot be sent by itself but only as a part of an electronic document or message. It is similar to an electronic “fingerprint". In the form of a coded message, the digital signature securely associates a signer with a document in a recorded transaction. Digital signatures use a standard, accepted format, called Public Key Infrastructure (PKI), to provide the highest levels of security and universal acceptance. They are a specific signature technology implementation of electronic signature (eSignature).

**4.2.5.2.1 COMPONENTS OF A DIGITAL SIGNATURE:**

A digital signature is made up of four components:

1. SHA-256 hash, which is a type of hash function.
2. Public Key.
3. Private Key.
4. Timestamp lists the precise time the certificate was issued.

**4.2.5.2.2 PUBLIC KEY INFRASTRUCTURES :**

In public key infrastructures, trusted bodies known as certification authorities, centrally manage the system by:

* Issuing the linked private and public keys.
* Running a server to timestamp each signature.
* Running the verification software.

Usually, the certification-authority embeds the public key in a certificate that contains a set of additional meta-data to facilitate usage. This offers several advantages:

* Certification authorities can verify the identity of persons to whom keys are issued, thus linking public-keys to real-world identities.
* Everyone can have confidence of the date of signature, since the ‘clock’ is maintained only by the certification authority.

However, public-key infrastructures also create a central-point of control and failure. Most critically, should the certification authority hosting the verification software close down (say, due to bankruptcy, civil unrest, restructuring etc.), it would effectively invalidate any document signed through it. This provides a significant problem for certificates such as birth, marriage or educational achievement which should last a lifetime.

If a private key is leaked, there is nothing to prevent an attacker from issuing fake records and backdating content. Even if an issuer publicly revokes those records, an independent verifier would not know the difference between a valid and invalid record unless there were some additional authority attesting to when the transaction took place.

**4.2.5.2.2.1 PUBLIC KEY:**

Public Key means the public address where other wallets send transaction values.

**4.2.5.2.2.2 PRIVATE KEY:**

Private Key means an encryption key uniquely linked to the owner and known only to the parties exchanged in a transaction; it is secretly held in a digital wallet.

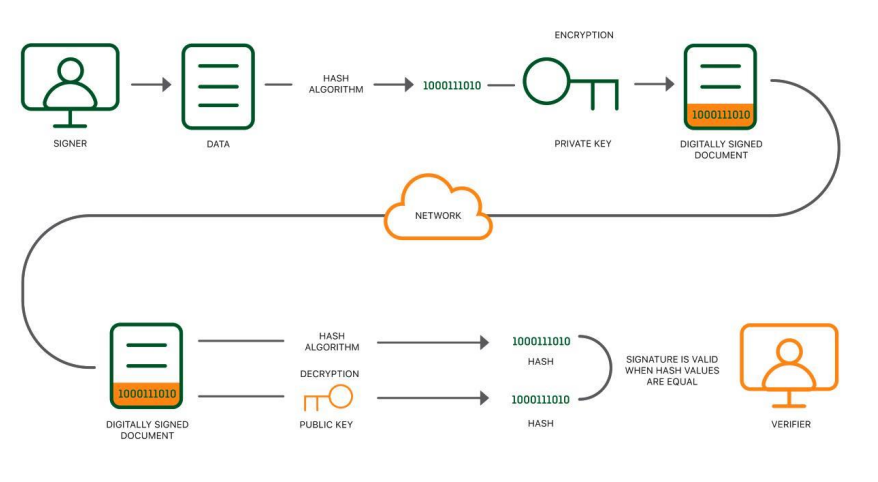


Figure 4.4: Anatomy of a Digitally Signed Document.

**4.2.5.3 HASHING:**

A hash is a short code of defined length which serves as a fingerprint for a digital document. A program called a hash-generator allows a user to upload any string of text and create a unique ID. Every time the same string of text is run through the hash generator, it will give the same document-ID. The contribution of hashing as an antitampering device is significant: if a single letter in a document is changed, it will automatically generate a completely different ID. Hashes are one-way, this means that the hash-generator can be used to generate a hash from the document, but it is mathematically impossible to generate a document from a hash.

A hash is the output of a hash function that expects an input value - in this case, PDF documents - and generates an output value in the form of a string of fixed length. The main feature of hash functions is that it is almost impossible to find two different input values that generate the same hash value. The hash function used in this approach is SHA-3 with a length of 256 bits. SHA-3, unlike MD5, is considered collision resistant, which means that the chance that two different input values produce different output values is very high. Hashes can be used to prove the authenticity of software artifacts. In this case, one speaks of checksums. To inform a user about the authenticity of downloaded software, companies often highlight the checksum on their website. The software can also generate a checksum which has to match with the checksum from the website. The checksum functionality can also be used for diplomas. If someone makes even the smallest changes to the document, the hash will change completely. SHA-3 is a one-way function, which means that it is not possible to recreate the input from the output. This property and also the uniqueness of hashes make it possible to encrypt diplomas without revealing confidential information. No one can interpret the 18 content of the diploma with the resulting hash, but it can be regarded as a unique link pointing to the official certificate.



Figure 4.5: Cryptographic Hash Function.

**4.2.6 BLOCKCHAIN PLATFORMS:**

There are several ways to create a blockchain, the most important of which are:

1. Bitcoin:

Bitcoin is an electronic payment system based on cryptographic proof instead of trust, allowing any two willing parties to transact directly with each other without the need for a trusted third-party. Invented by the pseudonymous Satoshi Nakomoto, it is the first implementation of Blockchain technology, and today the Bitcoin network still forms the largest public Blockchain in existence.

Bitcoin is an online equivalent of cash. Cash is authenticated by its physical appearance and characteristics, and in the case of banknotes by serial numbers and other security devices. However, in the case of cash there is no ledger that records transactions and there is a problem with forgeries of both coins and notes. In the case of Bitcoins, the ledger of transactions ensures their authenticity. Both coins and Bitcoins need to be stored securely in real or virtual wallets respectively — and if these are not looked after properly, both coins and Bitcoins can be stolen.

Due to a feature that allows it to store strings of up to 80 characters with every transaction, the Bitcoin blockchain is also being used as a public register to store hashes of documents. This in turn enables tamper-proof digital signatures.

Bitcoin is a fully open source project, and as such is governed by the community of Bitcoin users. Updates to the Bitcoin software, protocol and blockchain are accepted when more than half of the computers on the network choose to switch to a new version of the software. There are some limitations of the Bitcoin blockchain:

* It can only store the sender, receiver, amount of cash transferred and a hash.
* It can only process fewer than 10 transactions per second (compared to tens of thousands for a typical credit card network), a limit which has already been reached.
* Its size is growing exponentially, leading to a situation where only users with massive amounts of computing power can keep a copy of the entire Blockchain, reducing the number of computers in the network, and decreasing security overall.

2. Ethereum:

Ethereum is a blockchain platform and distributed computing system. It was released in 2015 and it is fully open-source. The main feature of Ethereum is its smart-contracts. The Solidity language is used to write said contracts using a high level language. They are compiled to bytecode that runs in the Ethereum Virtual Machine (EVM).

The advent of Ethereum created a new paradigm in a still-young blockchain industry and shifted its focus away from cryptocurrencies as financial tools and toward a more utilitarian purpose. With smart contracts on Ethereum and similar blockchains, processes that involve some transaction of data can achieve autonomy while remaining irrefutable and transparent. Startups and mature firms alike have developed ways to use smart contracts to build low-overhead work flows, and creatives are using them in their innovations as well.

Furthermore, Ethereum can process more transactions per second, and is more flexible in the amount and kinds of data which can be stored on it. In our project we will create our own blockchain based on the Ethereum network.

3. Hyperledger Fabric:

Hyperledger Fabric is an innovative project started at Linux Foundation, nowled and managed by two companies: IBM and Digital Asset. Hyperledger Fabric aims at providing a resilient, flexible, and confidential blockchain framework. It is considered the foundation of private, open-source blockchain applied to business.

Fabric’s architecture is far more complex than any blockchain platform while also being less secure against tampering and attacks. You would think that a “private” blockchain would at least offer scalability and performance, but Fabric fails here as well. Simply put, pilots built on Fabric will face a complex and insecure deployment that won’t be able to scale with their businesses.

**4.3** **ETHEREUM BLOCKCHAIN:**

Ethereum is a decentralized, open-source blockchain with smart contract functionality.

**4.3.1 SMART CONTRACT:**

Smart contracts are effectively small computer programmes stored on a blockchain, which will perform a transaction under specified conditions. Thus, a smart contract is typically a declaration such as “transfer X to Y if Z occurs”. Unlike a regular contract where after reaching an agreement, parties must execute the contract for it to take place, a smart contract is self-executing - that is, once the instructions are written to a blockchain, the transaction will take place automatically when the appropriate conditions are detected.

The objectives of smart contracts are the reduction of need in trusted intermediators, arbitrations and enforcement costs, fraud losses, as well as the reduction of malicious and accidental exceptions.

The Ethereum network is the most famous example of a smart contract based blockchain. It provides a virtual machine that executes, and charges for it, code on every single node of the network.

**4.3.1.1 EVM:**

The Ethereum Virtual Machine or EVM is the runtime environment for smart contracts in Ethereum. It is not only sandboxed but actually completely isolated, which means that code running inside the EVM has no access to network, filesystem or other processes. Smart contracts even have limited access to other smart contracts.

**4.3.1.2 SOLIDITY:**

Solidity is an object-oriented, high-level programming language for implementing smart contracts on various blockchain platforms, most notably, Ethereum.

It was influenced by C++, Python and JavaScript and is designed to target the Ethereum Virtual Machine (EVM)[S]. Solidity is statically typed. It supports complex member variables for contracts, including arbitrarily hierarchical mappings and structs. Solidity contracts support inheritance, including multiple inheritance with C3 linearization. Solidity uses ECMAScript-like syntax which makes it familiar for existing web developers;[citation needed] however unlike ECMAScript it has static typing and variadic return types. Solidity introduces an application binary interface (ABI) that facilitates multiple type-safe functions within a single contract.

**4.3.1.2.1 ABI:**

In computer software, an application binary interface (ABI) is an interface between two binary program modules. Often, one of these modules is a library or operating system facility, and the other is a program that is being run by a user.

An ABI defines how data structures or computational routines are accessed in machine code, which is a low-level, hardware-dependent format. In contrast, an API defines this access in source code, which is a relatively high-level, hardware independent, often human-readable format. A common aspect of an ABI is the calling convention, which determines how data is provided as input to, or read as output from, computational routines. Examples of this are the x86 calling conventions.

Adhering to an ABI (which may or may not be officially standardized) is usually the job of a compiler, operating system, or library author. However, an application programmer may have to deal with an ABI directly when writing a program in a mix of programming languages, or even compiling a program written in the same language with different compilers.

**4.3.1.3 IDE:**

There are many smart contract IDEs, the most common:

**4.3.1.3.1 REMIX:**

Remix is a web browser based IDE that allows to write Solidity smart contracts, then deploy and run the smart contracts. The best way to try out Solidity right now is using Remix.

**4.3.1.4 GANACHE:**

Another IDE for running smart contract. Ganache is a personal blockchain for rapid Ethereum distributed application development. You can use Ganache across the entire development cycle; enabling you to develop, deploy, inspect state and test your dApps in a safe and deterministic environment while controlling how the chain operates. All versions of Ganache are available for Windows, Mac, and Linux.

**4.3.1.5 TRUFFLE:**

A world class development environment, testing framework and asset pipeline for blockchains using the Ethereum Virtual Machine (EVM), aiming to make life as a developer easier. With Truffle, you get:

* Built-in smart contract compilation, linking, deployment and binary management.
* Automated contract testing for rapid development.
* Scriptable, extensible deployment & migrations framework.
* Network management for deploying to any number of public & private networks.
* Package management with Eth PM & NPM, using the ERC190 standard.
* Interactive console for direct contract communication.
* Configurable build pipeline with support for tight integration.
* External script runner that executes scripts within a Truffle environment.

**4.3.2 ETHEREUM WALLET:**

Ethereum wallets are applications that let you interact with your Ethereum account. Think of it like an internet banking app – without the bank. Your wallet lets you read your balance, send transactions and connect to applications.

Your wallet is only a tool for managing your Ethereum account. That means you can swap wallet providers at any time. Many wallets also let you manage several Ethereum accounts from one application. That's because wallets don't have custody of your funds, you do. They're just a tool for managing what's really yours.

**4.3.2.1 METAMASK WALLET:**

MetaMask allows users to store and manage account keys, broadcast transactions, send and receive Ethereum-based cryptocurrencies and tokens, and securely connect to decentralized applications through a compatible web browser or the mobile app's built-in browser. Developers achieve a connection between Metamask and their decentralized applications by using a JavaScript plugin such as Web3js or Ethers to define interactions between Metamask and Smart Contracts.

The Metamask application includes an integrated service for exchanging Ethereum tokens by aggregating several decentralized exchanges (DEXs) to find the best exchange rate. This feature, branded as MetaMask Swaps, charges a service fee of 0.875% of the transaction amount.

**4.3.3** **WEB3.JS:**

Web3.js is a library which provides functionalities to send transactions from a client to the Ethereum blockchain using HTTP, IPC or WebSocket.

**4.4 POLYGON NETWORK:**

Polygon is a cryptocurrency, with the symbol MATIC, and also a technology platform that enables blockchain networks to connect and scale. Polygon— "Ethereum's internet of blockchains"— launched under the name Matic Network in 2017. The Polygon platform operates using the Ethereum blockchain, and connects Ethereum-based projects. Using the Polygon platform can increase the flexibility, 28 scalability, and sovereignty of a blockchain project while still affording the security, interoperability, and structural benefits of the Ethereum blockchain.

**4.5 IPFS:**

IPFS is a peer-to-peer (P2P) distributed system for storing and accessing files. IPFS creates a hash of every single file stored in it. The files are, subsequently, accessed using these same hashes. Besides, it also features file versioning and duplicate file removal. Its uses are mostly for creating distributed file-sharing services. It is also widely used coupled with the Ethereum blockchain.

**4.6 PROOF OF EXISTENCE:**

The underlying concept of verification by blockchain technologies is called Proof of Existence. The principle behind this is that the proof of the existence of a document can be published anonymously and securely online. The service stores the cryptographic hash of the file. It is essential with this concept that the actual document is not stored or published anywhere under any circumstances. Therefore, the user does not have to worry about private matters to protect his information.

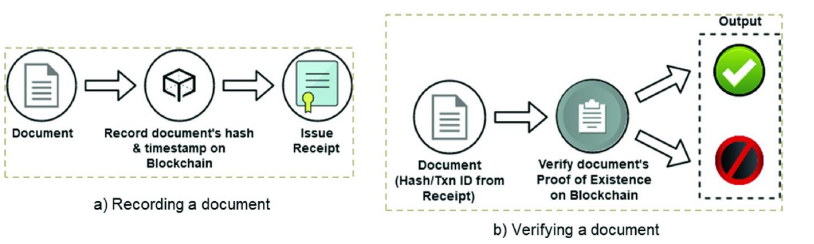
****

Figure 4.6: Recording and verifying documents to/from blockchain.

**CHAPTER 5**

**MODULE DESCRIPTION**

**5.1 UI DESIGN:**

* The UI Design consists of five pages – Home, Upload, Verify, Delete, Admin.
* Home is the index page of system which the user visits first.
* Upload is the page to upload the document.
* Verify is the page to verify the authenticity of the document.
* Delete page is to delete the document.
* Admin page is to control the overall system.

**5.2 ADMIN PANEL:**

* The person who integrates the system with the blockchain has the Super Admin access.
* Super Admin can add sub admins through their public key and the exporter name.
* Admin has the rights to add, edit and delete the exporter(sub-admin).
* With the public key, the sub-admin can upload the documents.

**5.3 UPLOAD PAGE:**

* The upload page is dedicated to the process of exporting documents by the officials.
* The process of exporting documents is done by selecting the document as a file through the “Choose File” entry, after which the document will be hashed directly.
* Once you click on the “Upload Document" button, it will wait for confirmation or rejection of the transaction.
* If the transaction is confirmed by pressing the "Confirm" button, the process of exporting the document will be completed successfully.

**5.4 VERIFICATION:**

* Organizations need to validate the applicant’s document, so they will visit the official website of the document provider for the purpose of verification.
* Once the document file is uploaded (or through a QR code), the document will be hashed.
* When the “verify” button is clicked, the system checks the document hash with the previously archived document hashes in the blockchain.
* If it matches, the document is valid otherwise the document is invalid.

**5.5 DELETION:**

* The exporters have the authority to remove a previously exported document.
* Through the “Choose File” entry, the document file is uploaded. Once the document file is uploaded, the document will be hashed.
* When the “Delete Document" button is clicked, it will wait for confirmation or rejection of the transaction.
* If the transaction is confirmed by pressing the "Confirm" button, the process of deleting the document will be completed successfully.

**CHAPTER 6**

**SYSTEM SPECIFICATION**

**6.1 HARDWARE REQUIREMENTS**

* Processor – Intel i3 or Higher.
* RAM – 4GB or Higher.
* Storage – 150GB or Higher.

**6.2 SOFTWARE REQUIREMENTS**

* Operating System – Windows 7 or Higher.
* Languages Used – HTML, CSS, Solidity.
* Blockchain – Ethereum.
* Tool – Metamask.
* JavaScript Library – Web3.
* Database – IPFS.

**CHAPTER 7**

**METHODOLOGY**

**7.1 INTRODUCTION**

To begin, in order to create a blockchain, we must create and deploy a smart contract. The Front-end will be constructed based on the created smart contract. Then, using Web3, we make the smart contract and the frontend communicate with one another. The transactions are then completed through the use of a digital currency wallet. The workflow is depicted in the diagram below:

**A diagram of a blockchain

Description automatically generated with low confidence**

Figure 7.1: Workflow of the implementation.

**7.2 SYSTEM ARCHITECTURE**

The system architecture and workflow mechanism are depicted in the figure below. The university that issued the document and the graduating student, as well as the company or the destination intended by the student for his own purposes, such as finding employment or completing postgraduate studies, are all relevant participants of the system. The university stores the documents of graduating students in its own archiving system (Blockchain) after graduation, and the student is provided a PDF file or a QR code for further verification. When a student decides to apply to a company in order to get a job or complete postgraduate studies at a university, he/she offers the institution his/her PDF or QR code file. To check the graduate student's priorities, this institution goes to the website of the university that issued the student's document. The validity of the document will be verified or refused by the university that issued it.

A diagram of a blockchain

Description automatically generated with medium confidenceFigure 7.2: Architecture and workflow mechanism of the system.

**7.3 IMPLEMENTATION DETAILS**

Explanation about the workflow in Figure 7.1 in more detail:

**7.3.1 SMART CONTRACT**

The Ethereum blockchain's core and brain is the smart contract. Remix is used to generate the smart contract, which allows you to develop Solidity smart contracts, then run and deploy them. Right now, Remix is the finest Ethereum IDE for trying out Solidity smart contracts.

Our smart contract is multi-functional:

* Adding and editing document exporters; the smart contract's owner (University) can add and edit a document exporter that represents one of the university's colleges. When the document exporter exports his college students' documents to the blockchain, this function is invoked.
* Delete a document exporter; the smart contract's owner (University) has the authority to delete a document exporter. The exporter will be unable to upload documents to the blockchain if the owner deletes the exporter.
* Upload a document; this function is only accessible to exporters, as it is used when each exporter from each college in the university uploads the documents of his college students to the blockchain and offers the student a copy of the document or QR code.
* Verification of documents; this function verifies the validity of documents issued by this university. When the verifier requests that a document be validated, this function is invoked.
* Delete a document; the owner or one of the exporters has the authority to delete a previously exported document.

Reviewing the smart contract code written in the Solidity programming language on Remix:

A screenshot of a computer program

Description automatically generated with medium confidence

A screenshot of a computer program

Description automatically generated with medium confidence

As aforementioned, the smart contract incorporates a number of functionalities that are designed to work as a system for issuing and verifying academic documents.

Explanation about each function in the smart contract:

Because it is impossible to manage an integrated system for issuing and verifying academic documents for a particular university by one person (Owner), multiple people known as exporters of academic documents must be involved. Each college had a designated document exporter for students graduating from that college. As a result, the smart contract's owner has control over two functions for this purpose: adding an exporter and editing its information, as well as deleting an exporter.

Function of adding and editing a document exporter; the smart contract's owner (University) can add and edit a document exporter that represents one of the university's colleges. When the document exporter exports his college students' documents to the blockchain, this function is invoked. In more detail, when the smart contract's owner (University) decides to add an employee from each college to act as a document exporter for graduating students, the system examines that the person doing the process is the owner. When the owner enters the address of an employee to be added as exporter, the system checks to see if the address is real first, then checks to see if the employee has already been added as exporter to the system. If these conditions are met, the process of adding a document exporter for a specific college is completed successfully. The flowchart below illustrates this function:

A diagram of a flowchart

Description automatically generated with low confidence

Figure 7.3: Flowchart of function of adding a documents exporter.

As we know that the administrative staff can be changed, so to enable the owner to manage the system more effectively, the function of deleting a document exporter from one of the colleges that are included in the document export system has been included.

Function of deletion a document exporter; the smart contract's owner (University) has the authority to delete a document exporter. The exporter will be unable to upload documents to the blockchain if the owner deletes the exporter. In more detail, when the smart contract's owner (University) decides to delete an employee from college to prevent him to act as a document exporter for graduating students, the system examines that the person doing the process is the owner. When the owner enters the address of the exporter to be deleted, the system checks to see if the address is actually exists as a exporter. If these conditions are met, the process of deletion a document exporter for a specific college is completed successfully. The flowchart below illustrates this function:

A picture containing text, design

Description automatically generatedFigure 7.4: Flowchart of function of deleting a documents exporter.

The university issues a document for each graduating student. The export process is restricted to document exporters exclusively. Each document exporter issues the documents of its graduating college students.

In more detail, when one of the exporters decide to issue a document for graduating students, the system examines that the person doing the process is one of the exporter. The system hashes the document when the exporter submits it. While the document is being uploaded to IPFS, an attempt is being made to upload the document hash to the blockchain. Then there's the waiting for transaction confirmation. If the transaction is confirmed, the next step is to wait for it to be included in the blockchain, so that the document's hash is stored on the blockchain while the document itself is stored in the IPFS. If these steps are done, the process of upload a document is completed successfully. The graduating student is given the hash of his document represented by a QR code in addition to the document as a file. The flowchart below illustrates this function:

A diagram of a flowchart

Description automatically generated with low confidenceFigure 7.5: Flowchart of function of export a documents.

When a graduating student desires to seek for a certain career or pursue postgraduate studies, he will submit the document file or QR code given by his primary university to the desired company or university.

The student's intended organization will go to the student's principal university website to see if the student is trustworthy or forger. This organization uploads the student document file (QR code can also be used) to the graduate student document verification page on the university's website. The system hashes the document in order to compare it to the hashes already included in the blockchain by the issuers. If the hash of the document given by the student is found in the blockchain, the next step is to go to the IPFS to view the document itself for more credibility. The process of verifying a document is successfully accomplished if these steps are met. Aside from that, the student is a forger and does not hold a degree from this university. The flowchart below illustrates this function:

A picture containing design

Description automatically generatedFigure 7.6: Flowchart of function of verify a documents.

To complement the functions of the system, we must include the function of deleting a document previously included in the system's database. In educational institutions, sometimes a document is withdrawn from a person, or an error is detected that requires correction later. Therefore, the exporters has the authority to delete a document that was previously exported.

In more detail, when one of the exporters decide to issue a document for graduating students, the system examines that the person doing the process is one of the exporter. The system hashes the document when the exporter submits it. The system hashes the document in order to compare it to the hashes already included in the blockchain by the issuers. If the hash of the document is found in the blockchain, the intended document will be deleted. If these steps are done, the process of upload a document is completed successfully.

This process entails the consequences of not being able to prove the authenticity of the document of the person who is the target of the deletion process. The flowchart below illustrates this function:

A diagram of a flowchart

Description automatically generated with low confidenceFigure 7.7: Flowchart of function of deleting a documents.

**CHAPTER 8**

**CONCLUSION AND FUTURE ENHANCEMENT**

**8.1 CONCLUSION**

This system automates the process of generating Certificates and reduces the manual work needed for the verification of the same. Students are also at a comparatively low risk of losing the certificate. By using an additional hashing algorithm, it decreasing the percentage of data being tampered with. The SHA hash function has so far proven to be safe, which means that no two different inputs resulted in the same output. At the same time, it is not possible to generate real information from a hash, where the same document cannot be retrieved by the produced hash. Retrieval of the same document is only done by IPFS.

**8.2 FUTURE ENHANCEMENT**

To further enhance the capability of this project, we recommend the following features to be incorporated into the system:

* To store document files on servers located throughout the region of interest.
* Provide better user interface for user.

**APPENDIX A**

**Sample code**

**index.html**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta name="viewport content="width=device-width, initial-scale=1,

shrink-to- fit=no"/>

<!-- SEO Meta Tags -->

<meta name="description" content="Your description" />

<meta name="author" content="Your name" />

<meta property="og:site\_name" content="Certify" />

<!-- website name

<meta property="og:site" content="certify.com" />-->

<!-- website link -->

<meta property="og:title" content="Certify" />

<!-- title shown in the actual shared post -->

<meta property="og:description" content="Certify Is A Website For institutions To Verify The Authenticity Of Your Documents Online Easily And Safely. Let's Go.." />

<!-- description shown in the actual shared post -->

<meta property="og:url" content="#ABOUT" />

<!-- where do you want your post to link to -->

<meta name="twitter:card" content="summary\_large\_image" />

<!-- to have large image post format in Twitter -->

<!-- Webpage Title -->

<title>Certify</title>

<link rel="stylesheet" href="./css/loader.css">

<link href="./css/bootstrap.min.css" rel="stylesheet" />

<link href="./fontawesome-free-6.1.1-web/css/all.min.css" rel="stylesheet" />

<link href="./css/aos.min.css" rel="stylesheet" />

<link href="./css/main.css" rel="stylesheet" />

<!-- Favicon -->

<link rel="icon" href="./assets/images/icon.png" />

</head>

<body>

<div class="loader-wraper">

<div class="ldsroller"><div>

</div><div>

</div><div></div>

<div></div><div>

</div><div></div><div></div><div></div></div>

</div>

<nav class="navbar navbar-expand-lg navbar-light bg-light py-3 navbar-dark">

<div class="container">

<a class="navbar-brand" href="index.html">

<i class="fa-solid fa-dragon home\_text"></i>

<span class="home\_text">Certify</span>

</a>

<button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarSupportedContent">

<ul class="navbar-nav me-auto mb-2 mb-lg-0">

<li class="nav-item">

<a class="nav-link active" aria-current="page" href="index.html">Home</a>

</li>

<li class="nav-item">

<a class="nav-link " href="upload.html">Upload </a>

</li>

<li class="nav-item">

<a class="nav-link" href="verify.html">Verify </a>

</li>

<li class="nav-item">

<a class="nav-link" href="delete.html">Delete </a>

</li>

<li class="nav-item">

<a class="nav-link" href="admin.html">Admin</a>

</li>

</ul>

</div>

</div>

</nav>

<!---------------------NavBar End-------------------->

<section class="home py-5 d-flex align-items-center" id="header">

<video autoplay muted loop id="myVideo">

<source src="./files/earth1.mp4" type="video/mp4">

</video>

<div class="container text text-light py-5" data-aos="fade-right">

<h1 class="headline text">Build <span class="text-info">trust</span> <br>into your Organization</h1>

<p class="para py-3">

The Certify platform is a leading Document Verification platform <br>

designed to bring efficiency and security to your operations.

</p>

<div class="my-3">

<a class="btn bg-danger text-white" href="verify.html">Go Verify</a>

</div>

</div>

</section>

<!-- End Footer -->

<div >

<i onclick="topFunction()" id="scroll-btn" class="fa-solid fa-angle-up"></i>

</div> <div>

<script src="https://code.jquery.com/jquery-3.6.0.js"integrity="sha256-H+K7U5CnXl1h5ywQfKtSj8PCmoN9aaq30gDh27Xc0jk=" crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"></script>

<!-- Bootstrap framework -->

<script src="./js/purecounter.min.js"></script>

<script src="./js/aos.js"></script>

<script src="./js/script.js"></script>

<script defer>

$(".loader-wraper").fadeOut("slow");

</script>

</body>

</html>

**APPENDIX B**

**SAMPLE OUTPUT**

**MODULE 1: UI DESIGN**

**A computer screen capture

Description automatically generated with medium confidence**

**MODULE 2: ADMIN PANEL**

**A screenshot of a computer

Description automatically generated with medium confidence**

**MODULE 3: UPLOAD PAGE**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**MODULE 4: VERIFICATION**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**MODULE 5: DELETION**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**REFERENCES**

1. Kumutha.K and S.Jayalakshmi, "The Impact of the Blockchain on Academic Certificate Verification System-Review", EAI Endorsed Transactions on Energy Web, 2021.
2. Geetha S K, and Sreenithi R, Assistant Professor, Dept. of CSE, KPR Institute of Engineering and Technology, Coimbatore, Tamil Nadu, India, “Educational Certificate Verification Using Blockchain Based Framework”, Turkish Journal of Physiotherapy and Rehabilitation, 2020.
3. "Secure Document Verification using Blockchain Technology" by R. D. Gawande, V. N. Nemade, and S. S. Suryawanshi (2019).
4. "Blockchain-Based Authentication and Verification of College Degrees" by K. Bhattacharya and S. Saha (2019).
5. "Blockchain-Based Document Verification for Ensuring Data Integrity" by N. Parveen, A. T. Ngo, and H. T. Nguyen (2018).