**Cloud storage verifier using Block-Chain**

**Student ID:**

**Student Name:**

# Abstract

Cloud computing, as a network-enabled technology, has been widely adopted in the business world through a number of cloud service models. In a variety of ways, blockchain technology has been touted as a good choice for reinforcing existing computer systems. Combining blockchain technology with existing cloud services has enormous promise in terms of functionality, performance, and security/privacy. (Gai et al., 2020).

With businesses looking for faster, more nimble solutions to their web development needs, it's no wonder that there has been a proliferation of abstract flask web development API clouds over recent years. Collaborative flask blockchain cloud is a new concept that enables organizations to manage, monitor, and store data securely. The collaborative flask blockchain cloud provides an efficient platform for sharing files, documents, and other types of information between multiple users. It also allows the organization to analyze data quickly and make decisions based on this analysis. Nowadays Websites use blockchain technology to process files. It is an extremely secure and reliable way to handle files. Collaboration with the cloud also provides an incredible speed and another layer of security for any application. The blockchain technology uses cryptography to ensure that files are not tampered with or copied illegally. It also ensures that the original file is always available and can be verified.

Keywords—*Blockchain, Cloud, Python, Flask, Web Development, Truffle framework, Ganache, Smart contract*

**Table of Contents**

[Contents](#_heading=h.1t3h5sf)

[Abstract 1](#_Toc103959741)

[1. Introduction 3](#_Toc103959742)

[1.1 Research background 5](#_Toc103959743)

[1.2 Aim 6](#_Toc103959744)

[1.3 objectives 6](#_Toc103959745)

[2. literature survey 6](#_Toc103959746)

[2.1 Technical Challenges 9](#_Toc103959747)

[3. Requirements & Tools 9](#_Toc103959748)

[3.1 Blockchain 9](#_Toc103959749)

[3.2 Truffle 11](#_Toc103959750)

[3.3 Ganache 11](#_Toc103959751)

[3.4 Python 12](#_Toc103959752)

[3.5 Flask framework 13](#_Toc103959753)

[3.6 Firebase 14](#_Toc103959754)

[3.7 Bootstrap 15](#_Toc103959755)

[4. Design 16](#_Toc103959756)

[5. Installations 17](#_Toc103959757)

[6. Implementation & Approach 17](#_Toc103959758)

[7. Testing 20](#_Toc103959759)

[8. Evaluation and Results 22](#_Toc103959760)

[9. Conclusion 26](#_Toc103959761)

[10. Bibliography 26](#_Toc103959762)

# Introduction

Blockchain technology enables secure transactions amongst untrustworthy network participants (Aste et al., 2017). Because of its unique trust and security properties, A growing number of scientific research have been focused on blockchain technology. and has aroused the curiosity of academics, technologists, and industry practitioners. Since the publishing of the first Bitcoin blockchain in 2008, several blockchain systems, such as Hyperledger Fabric and Ethereum, have emerged with the assistance of private and public accessibility outside of traditional fiat currencies and electronic voucher systems. (Taylor et al., 2020)

There is no doubt that blockchain has become increasingly popular around the world. Rather than simply being renowned, it has had a lasting impact on the world. It has been economically accepted, influenced global money, and aided the expansion of illicit dark web markets, for example. It has also aided the proliferation of financially motivated cyber-attacks on merchants and other online enterprises, such as ransomware and denial-of-service attacks. The implementation and use of blockchain has far exceeded its original intent as the foundation for the world's first decentralised cryptocurrency. Other industries have recognised the benefits of a decentralised ledger with historical immutability and are working to apply the core concepts to their current business operations. Blockchain technology's unique characteristics make it interesting for a wide range of business domains and industries, including banking, logistics, the pharmaceutical industry, smart contracts, and, most importantly, cyber security. (Taylor et al., 2020)

Due to one-way cryptographic hash algorithms and community consensus, blockchain is a distributed, tamper-proof mirrored ledger where records are unchangeable. Immutability eliminates the requirement for reconciliations, resulting in a unique historical reconciliation version of reality. The generation of confidence in the system is a direct result of an immutable historical record certified by community consensus. Indeed, tampering with such a record becomes extremely difficult for a person or a group of individuals. (Aste et al., 2017)

One of the most promising communication and information technologies (ICT), cloud computing, is an effective way to possibly overcome the constraints of IIoT nodes. It also frees consumers and businesses from high hardware expenditures and low productivity as a paradigm for giving on-demand network access to customizable computer resources. Following that, the rapid growth of the global cloud market has been fueled by the spread of commercial and research cloud applications. In addition, revenue from cloud services is predicted to exceed 300 billion dollars by 2021. (Xie et al., 2020)

Cloud services are often purchased from companies like Google Cloud Platform, Microsoft Azure, Amazon Web Services (AWS), and Alibaba Cloud. These cloud behemoths have their own cloud markets, making it difficult for customers to select relevant and cost-effective services. Due to the lack of competition, these providers typically charge a greater price for their cloud services than they would in a competitive market. Meanwhile, a single cloud-service provider is unable to meet the demands of installing many various cloud services for a global firm. (Xie et al., 2020)

A smart contract is a programme that runs on a blockchain and it also stores data in it. It is a database which is globally replicated. A smart contract's constructor and methods encapsulate the business logic of a distributed web application known as a Distributed Application (DApp). The function constructor and function calls in the DApp generate one or more transactions that are recorded in a blockchain block. The blockchain is simply a replicated append-only database. The Ethereum blockchain is administered by miners, who operate a worldwide peer-to-peer network. Miners employ a distributed agreement procedure to decide which blocks are put to the blockchain. The miners are compensated monetarily for their work. Transactions on the blockchain are generally persistent after they are saved in a block (except in the case of forks). Transactions are censorship resistant due to the blockchain's global replication. (Hartel & van Staalduinen, 2019)

The Ethereum Virtual Machine (EVM) byte code created by compiling the contract source is also saved in the blockchain when a contract is deployed. The Ethereum blockchain not only stores contracts and transactions, but it also keeps track of a monetary balance in Ether, the native currency of Ethereum. There are two kinds of accounts- first one is externally owned accounts, which are created by the establishment of a public private key pair, Contract accounts, on the other hand, are defined by the deployment of a contract. Every transaction does have its own hash and each contract or externally controlled account does have its own address. (Hartel & van Staalduinen, 2019)

A transaction might modify the account balance, and it normally sends out events to the governing DApp. A return value does not exist for a transaction. Each successful transaction necessitates the consumption of a particular amount of gas, which itself is proportional to the amount of work completed under the contract. As a result, ether and the overall gas consumption of a particular transaction are important contract outputs. A transaction has status because it may fail. The transaction will result in failure if the quantity of gas given is insufficient, for example. This method is used to prevent infinite loops, among other things. Only users with the correct private key can perform transactions that are cryptographically signed. (Hartel & van Staalduinen, 2019)

Pure functions in a contract can only access and read the state of the contract from the blockchain, as well as methods that change the state. Pure functions are used to send information about the state of the contract to the DApp. Pure function calls always succeed and cost no gas. The Truffle framework is frequently used by Ethereum smart contract developers to test their contracts. The tests are normally implemented using JavaScript, as well as the truffle framework ensures that once the contact is deployed on one of the prototype networks or the main Ethereum network, the tests can interact with it. (Hartel & van Staalduinen, 2019)

Python is a high-level generic programming language that is extremely popular. In software engineering,Since it is a dynamically typed language, it is known as duck typing. It is a language that is translated, as it does not require to be compiled separately, instead, it's compiled to bytecode and executed instantaneously. (Ghimire, 2020)

It is no longer just a scripting language, thanks to its object-oriented architecture. Beyond scripting, the language has advanced to the web and large data area. Frameworks like Flask, Django, and Tornado are available for web development. The frameworks listed above are also the go-to frameworks for constructing a decent web app in a short period of time. (Ghimire, 2020)

To quickly develop a web application, flask is a micro-framework which permits users to do it. It just implements the most basic functions,allowing developers to add additional features as needed throughout the development process. flask is a WSGI application framework that's compact and light. Both backend and frontend of an application can be developed using this framework. A complete request, objectAn interactive debugger, HTTP utilities for handling entity tags, and an endpoint routing system, dates, cache controls, and cookies are among the features included in the former. For local development, A threaded WSGI server and the 9 test client for simulating HTTP requests are also included with flask. (Ghimire, 2020)

Flask is known as a prototype framework since it lacks a database abstraction layer, as well as any form of validation or security. Flask has provided the implementer complete control over how the criteria are introduced as a consequence. Flask is compatible with Python 3 and later, also PyPy, and “Pip”, Python's official package manager, can be used to install it.(Ghimire, 2020)

Here, we are attempting to create a cloud-based programme that displays file upload and download while also incorporating block chain technology for an additional degree of security validation.

## 1.1 Research background

A brief summary of the most important studies is included in the research background. The backdrop of each investigation must be established. A solid background establishes context and encourages the reader to read the material quickly. The backdrop of the research differs from the literature review. So far, background research has been completed in chronological sequence. Background research is when the source of the problem is investigated. It identifies and explains a problem's nature and history. The paper's concentration is not on background research, but it is dependent on keeping the essential themes in the introduction. It identifies and explains a problem's nature and history. The paper's concentration is not on background research, but it is dependent on keeping the essential themes in the introduction. Background information in a document serves as a link between the reader and the subject. A backdrop is a method of analysis. This contains both important and relevant research from the publication. (Alfiyatin et al. p-323).“Digitalization has transformed our daily lives; therefore, it is very important for us to know if the information we need is 100% trustworthy.With blockchain verification, users can be certain that the data they are storing is accurate and secure,” said David Dickson, founder and CEO of Datonics.Cloud computing allows users or organizations to store data remotely rather than storing everything locally.As most businesses and institutions shift toward mobile devices, Cloud Computing helps customers manage content remotely. Blockchain creates public ledgers of transactions that cannot be tampered with or manipulated electronically. With blockchains, organizations can verify and track digital assets from creation through transmission. Cloudchain is a type of smart contract that maps the identities of cloud providers to their addresses on the blockchain. Similar contracts are proposed in the context of medical data management. A status variable is stored in each transaction list, indicating whether the transaction is pending and has been completed or not. Cloudchain participants can join and depart the system at any moment by performing particular smart contract procedures. To validate the service quality provided by the blockchain network, an oracle is necessary. (Taghavi et al., 2020) Cloud chain is a new type of secure storage technology that uses blockchain. Cloud chain stores data in multiple nodes and each node has the ability to access, store, and manage the data. The system also allows users to share files with other cloudchain users without having to use a third party service like Dropbox or Google Drive.Cloud chains provide an extra layer of security because all participants have full control over their data.Cloud chain works by creating a chain of blocks. Each block contains information about a particular file or data item. The blocks are joined together in a chain by the nodes in the network. The chain is designed to ensure that the data is always accessible and that it is protected from being tampered with.

Maintaining a clearer system is important for business success. This is because it allows employees to understand their tasks and work efficiently together. It also allows supervisors to keep employees informed and organized. This leads to a better work environment and improved communication. The cloud does not provide that much of transparency but with the help of block chain it does provide the access log of activities to every node in the system. This enables the fast monitoring of the transactions. Also, with the help of additional data the logs can be filtered very efficiently. Not only the security, the block chain acts as a backup of the cloud storage, as it can store encrypted data.

However, the cost of the implementation in the large scale can be a big factor, and the speed might be affected for the verification, furthermore the data retrieving can also be a hardship if ever the cloud fails and needs to be retrieved from the block chain.

Cloud is not a vulnerable storage and computing system. The popular cloud providers like google cloud service, AWS, Microsoft azure does provide multiple strong layers of security. However, the service providers provide a good firewall from the external sources for an instance but from inside of the instance the entities can create exploitation breeches.

## 1.2 Aim

The above surveys state that the insiders of an instance can cause a lot more problems. Not only the intense exploitation, times some fault transactions can happen in the cloud which cannot be tracked.

In this project my aim is to build a system with the Web interface that can demonstrate cloud activities and secure that system using a private block chain connected by a smart contract.

Also, the system will provide every end user to see the entire log of the transaction that has occurred. Here the users will be able to download and upload any file. and also will be able to delete the file from everyone in the system.

Summarizing my aim is to understand how the blockchain technology works and provide security to the cloud system.

## 1.3 objectives

In this project my main objective is to understand the usage of a blockchain to provide a layer of security over the cloud services, which in this project is represented by cloud storage (google cloud storage = firebase).

# literature survey

Web applications have grown in popularity in recent years, and they are now frequently used in security-critical domains such as medical, financial, and military systems. As the frequency and complexity of threats against essential services web apps has grown, so has the amount and cleverness of attacks against these apps. To detect and block web-based attacks, using vulnerability research software to find security flaws prior to deployment or existing practices to online application safety rely on application-level firewalls. (Balzarotti et al., 2007).Web applications are computer programmes that operate on the Internet. They come in a variety of shapes and sizes, but they all have one thing in common: they're all designed to be accessed through a web browser. Web application security is a subset of cybersecurity that addresses vulnerabilities that arise during the development, deployment, and use of web applications. It covers anything from preventing unwanted access to data contained in web applications via adequate authentication and permission restrictions to detecting and mitigating cross-site scripting (XSS) assaults on unwary visitors. Web application security has improved over time. Web applications were initially vulnerable to basic assaults. Security has, nevertheless, been a larger issue for online application developers in recent years.

In the year 2017, the Open Web Application Security Project (OWASP) issued a list of the most common web application vulnerabilities, explosion of sensitive data, including injection, broken authentication, security misconfiguration, External XML entities, cross site scripting (XSS), insecure deserializationm, broken access controls, insufficient logging and monitoring, and using component with known vulnerability. Misconfiguration and authoring program code are the two most typical blunders made when establishing a website application. Because of these two reasons, web-based apps are insecure. (Setiawan et al., 2020).

By exploiting errors in the construction of software code, Cross-site scripting and SQL injection attacks are possible (XSS). Cross-site scripting (XSS) is a sort of injection attack in which malicious script code is injected into an unsuspecting website visitor's session without their knowledge or consent, using the user's web browser. XSS is most commonly used to exploit weaknesses in websites that aren't properly secured. A hacker who successfully executes an XSS attack can gain access to sensitive information such as login passwords, session data, and more by exploiting the victim's confidence. Cross-site request forgery (CSRF), sometimes known as cross-site poisoning (CSP), is a type of injected material that makes users submit requests on behalf of attackers rather than themselves.

As a result, attackers may exploit the system to get access to sensitive data or information stored in the online application. According to Akamai data from November 2017 to March 2019, SQL injection attacks accounted for 65.1 percent of all web application attacks. Following that, attacks on input validation account for 24.7 percent of all attacks, while assaults on the type of Cross-Site Scripting (XSS) account for 4.5 percent. Testing the security of a system is one approach to assess and certify security. To establish whether or not a security test can be undertaken, a vulnerability assessment can be performed. Vulnerability assessments are performed on a regular basis to uncover system flaws. (Setiawan et al., 2020)

Blockchain technology is really exciting because it offers a high level of security. Blockchain is similar to a database in that it organises records into blocks in a ledger rather than maintaining them in a table. The name Blockchain comes from the fact that each new block is connected to the preceding block using cryptographic hashes. The record may be disseminated to all nodes in the network for verification and validation. The process of producing and certifying a block is referred as as "consensus." (Alketbi et al., 2018) Blockchain technology is really exciting because it offers a high level of security. Blockchain can be used to store any type of data, including medical records, legal documents and financial transactions. The unique feature of blockchain is that it allows users to access the data only if they have the correct password or ID. This makes it difficult for anyone else to access the data without your permission. Blockchain technology is a novel technique to facilitate transactions between two or more parties without requiring trust. A digital ledger is used to record transactions, which can then be validated by all relevant network participants. This lets people make transactions without the use of third-party intermediaries, which has significant security and transparency advantages. The blockchain transactions allow for the movement of value among two parties without the requirement for trust between the parties or a centralised authority. Because Blockchain transactions are programmable, they can hold any data relevant to the application. (Alketbi et al., 2018) The benefits of using the blockchain for transactions include speed, security, and transparency. Transactions on the blockchain are faster than traditional transactions due to the elimination of the need for a third party (such as a bank). Transactions on the blockchain are also more secure due to the fact that they are cryptographically secured using blockchain technology. Lastly, transactions on the blockchain are transparent, which means that everyone involved in a transaction can see it in real time.

The recipient's public address, transaction value, and digital signature should all be included in the transaction created by the sender. In transactions, digital signatures are vital since they assist guarantee that the transactions are genuine. Digital signatures help to ensure that the transaction is legitimate. This is because digital signatures help to ensure that the person who is sending the money is the person who is supposed to be sending the money. Digital signature is a security feature in computing systems that allows a user to certify the identity of another user. A digital signature can be used for two main purposes: authentication and integrity verification. Authentication uses digital signatures to verify the identity of an individual, while integrity verification verifies that information has not been altered since it was signed. Smart contracts are computer protocols designed to facilitate, verify, or enforce the negotiation or performance of a contract without third-party involvement. A private block chain is an encrypted database where transactions take place between parties who know each other's secret keys (or have agreed to do so). The message's authenticity is verified using the digital signature. This transaction is then sent to the network, where nodes must check the transaction's legality using the digital signature, and if it is, it is sent to the "unconfirmed/unordered transactions pool." Every authorised node in the network keeps a record of all transactions in Blockchain, and transactions are stored in blocks that create a ledger in the form of Block chain technology. (Alketbi et al., 2018)

A smart contract is a part of a software that is run on the blockchain system using the consensus mechanism. A smart contract work by using transaction codes and a blockchain. A transaction code is a set of rules that is used to execute the contract. The transaction codes are stored on the blockchain and verified by the network. If the code is verified, then the contract is executed. A smart contract is significant because it is a blockchain-based digital agreement. All smart contract transactions are recorded on the blockchain, which is a public ledger. The agreement can now be verified without the involvement of a third party. The network performs the verification process. This renders smart contracts untrustworthy. The blockchain also ensures the agreement's accuracy.a series of occurrences A smart contract can be utilised in a variety of fields.to remove third-party transactions and automate the process system. We identified seven different use cases in our research. displayed in a smart contract and blockchain-based application. The numerous types of blockchain applications already described technologies (Mohanta et al., 2018). Smart contracts are still in their infancy in terms of research. Blockchain is a peer-to-peer network that is not controlled by a central authority or server. Every single person or equipment in the network is referred to as a "node." A node machine can be a mining device, a storage device, or any other device that serves a purpose. Smart Contracts may be created on the blockchain. Smart contracts are bits of code that are stored on a blockchain network and execute the agreement terms automatically.

The cloud storages and nosql databases are not safe enough for this vulnerability issues of the website and unsecured transactions. For conquering this un-authorised transaction issues the blockchain technology provides the additional log files which helps to validate the cloud database transactions.Cloud storages are a popular way to store data. However, they are not safe enough for this vulnerability issue. Cloud storages are vulnerable to data leak due to unsecured transactions. This means that hackers can steal your data if they are able to get access to your cloud storage. In addition, cloud storage can be insecure because it is not always made from secure networks. This means that it may be possible for hackers to attack your cloud storage and steal your data.

The article overview of cloud storage provided me with the broad knowledge needed for this project, as well as many elements of it, such as how the old strategy of keeping data locally on the user's hard drive cannot keep up with changing user requirements. Users want data to be available at all times, on any device, from any place. For such customers, cloud storage becomes the magical answer. Every day, every organisation generates vast volumes of data, which is rapidly increasing. This has resulted in the consumerization of IT, resulting in a significant need for large amounts of storage that can be accessed from anywhere using any device. (Elzeiny et al., 2013)

The whole database is viewed as a single huge piece of data stored on Storage Area Network (SAN) or Network Attached Storage (NAS) storage that is shared and accessible across the network by all nodes in this design. The shared-nothing design is also used by Amazon's SimpleDB, Hadoop Distributed File System, and Yahoo's PNUTS. Higher data transportation equals more network capacity issues and delay. These problems have a significant impact on database performance. It's not as easy to dynamically add another database server as it is to spread the data across many servers. (Elzeiny et al., 2013)

## 2.1 Technical Challenges

The project is designed to connect to a number of other technologies. These technologies include cloud storage, private block chain, web-app. The project is designed to communicate with a private block chain technologies system. The project is designed to receive updates from the cloud-based system. Also is designed to send updates to the cloud-based system.

To begin, there are numerous cloud storage providers available to support any type of system; however, this project requires an appropriate cloud storage that can provide both storage and an authentication service. The google cloud was the most appropriate solution for this .

Next for the blockchain it was difficult to develop a block chain or use a public blockchain, thankfully the ganache provided a private block chain service at ease.

The connection with the cloud and blockchain needs a Smart contract which was developed with the help of truffle using solidity.

For the interface bootstrap framework and Tailwind css was able to provide a simple yet effective UI and also was easy to develop.

finally, for connecting everything together it was really a challenge to use only node Js. Thankfully the flask framework based on python connected everything. It was really powerful yet very easy to develop. It connected the web interface with the cloud storage for file transactions with the firebase admin credentials. Nextly it connected the block chain with the cloud with the smart contract.

# 3. Requirements & Tools

## 3.1 Blockchain

Blockchain technology was used for the first time when the Bitcoin cryptocurrency was launched. It is well known as the technology that underpins the Bitcoin cryptocurrency. Blockchain is a distributed ledger technology that maintains transaction data integrity. Bitcoin is a decentralised digital currency payment system based on the public transaction log Blockchain. Today, Bitcoin is the most commonly used Blockchain application. The most crucial feature of Bitcoin is that its value may be maintained without the involvement of any institution or government. The Bitcoin network is continually growing in terms of transactions and users. Furthermore, in currency exchange markets, common currency conversions such as EUR, USD, and KRW are conducted very often. As a consequence, Bitcoin has grown in popularity and is now the most widely used digital currency based on Blockchain technology. (Yli-Huumo et al., 2016)

Blockchain is a decentralised database system that maintains a continuous set of data transactions that are authenticated by the network's nodes. The purpose of Blockchain technology is to establish a decentralised environment in which transactions and data are not controlled by a third party. Blockchain is a distributed system that does away with the requirement for a middleman. The data is stored in a public ledger that contains every transaction ever made. The each and every transaction ever made in Blockchain is published and visible to all nodes. This feature enables the system's transparency when a third party is involved in centralised transactions. In short blockchain is a distributed database system that maintains a continuous set of data transactions. The purpose of the technology is to establish a decentralised environment in which transactions and data are not controlled by a third party. Each and every transaction in the system is published and visible to all nodes. Additionally, because all Blockchain nodes are anonymous, transaction confirmation is safer for other nodes. Bitcoin was the first application to use Blockchain technology. It built a distributed cryptocurrency ecosystem where in the users may use digital currency to purchase and sell goods. (Yli-Huumo et al., 2016)

Regardless of the fact that Blockchain technology appears to be a feasible choice for carrying out financial transactions, it will still have substantial technological challenges and limitations that must be investigated and addressed. Confidentiality and high transaction consistency, as well as node anonymity, are essential in Blockchain to avoid attacks and attempts to disrupt transactions. Likewise, validating transactions with in Blockchain requires some computer processing power. (Yli-Huumo et al., 2016)

The Cryptocurrency blockchain is a distributed management method allowing Bitcoin currency users to issue and move money. Without the intervention of a third party, this system can keep documentary evidence of all Cryptocurrencies that have ever been done. The benefit of Blockchain is that once data has been approved by all nodes, it cannot be changed or removed from the public ledger. This is why the data integrity and security properties of Blockchain are well-known. Other sorts of applications are possible using blockchain technology. It can, for instance, develop a cloud service that allows for digital contracts and peer-to-peer data sharing. Data integrity is a strong suit of the Blockchain method, which is why it's being used in different businesses and applications. (Yli-Huumo et al., 2016)

Overall, Blockchain has the potential to transform the way people conduct transactions in everyday life. Additionally, the technology's applications are not limited to cryptocurrencies; it might be employed in a variety of scenarios where specific types of transactions are carried out. Despite future research into the uses of Blockchain is likely to be interesting, it now confronts technological limits and challenges. The qualities of anonymity, data integrity, and security create a number of exciting challenges and problems that must be addressed and evaluated through high-quality research. Additional concern which must be overcome in order to satisfy future needs is scalability. As a result, in order to identify and appreciate the current status of Blockchain research, all relevant research must be gathered. Then you may assess what obstacles and concerns have been addressed and solved, as well as what the most pressing Blockchain issues are right now. (Yli-Huumo et al., 2016)

## 3.2 Truffle

Truffle Framework is a must-have tool for Ethereum smart contract developers. It creates smart contracts using the Solidity programming language. Truffle framework provides following functionalities:

* Truffle is a client-side framework, so dynamic applications have improved functionality and security.
* A plug-in Script Runner in compiler
* Network Management: Network management is the process of determining the condition of your network's operation. It also allows you to resolve a variety of network issues, both known and unknown.
* It provides Development Console for better customisability
* Deployment & Migrations: The migration and deployment process can be divided into five major phases,
  + Planning: During the planning phase, you review information such as assets, objects, hardware and software requirements, system and asset configuration, and other aspects of the source and destination environments.
  + Migrating: During the migration phase, you use information from the planning phase to prepare an application package for migration.
  + Analyzing: During the analysis phase, you analyze the impact of the deployment on the destination environment.
  + Backing up the system: During the backup phase, you suspend activity on the system for the destination environment and create a backup for that system.
  + Deploying: During the deployment phase, the sequence of individual tasks that are required to deploy an application into the destination environment can vary widely.
* Smart contract and smart contract management: A smart contract is a blockchain-powered decentralized application that self-executes itself when pre-defined ‘if/then’ conditions are met. Smart contracts are not managed by a central body or authority. Smart contracts can have varied applications across different enterprises. A smart contract can help facilitate service deliveries, money exchange, unlocking digital content protected by digital rights, changing the name or title of land, and more.
* Automated Testing: Automated Testing (also known as test automation) is a software development approach in which tools are used to execute specific tests on software in response to a trigger. These tests are frequently made up of procedures that are both repeatable and have a predictable outcome. (Bhosale et al., 2019)

## 3.3 Ganache

A private blockchain is a blockchain that may be utilised both publicly and privately. A private blockchain differs from a regular bank ledger, which is used by organisations or entities to keep track of transactions and financial data. Companies frequently employ private blockchains to develop their own digital tokens, smart contracts, and other network applications. *Ganache* is a tool for deploying smart contracts and testing them. You've got a personal blockchain network up and running. Ganache gives ten Ether accounts for testing its smart contracts on local blockchains. (Bhosale et al., 2019) It provides the various functionalities

## 3.4 Python

Python is an interpreted and interactive object-oriented programming language. High-level data structures like as lists and associative arrays (also known as dictionaries), dynamic binding and dynamic typing, classes, modules, automated memory management, exceptions, and so on are all included. It's a versatile programming language with a remarkably simple and beautiful syntax. (Sanner & others, 1999)

It is free, even for commercial usage, and, like many other scripting languages, can be used on practically any contemporary computer. A Python programme is automatically compiled into platform-independent byte code, which is then interpreted by the interpreter. Python components are operating unchanged on Linux, Windows 98, Windows NT, IRIX, 95, OSF, and SunOS. (Sanner & others, 1999)

Traditional static code analysis is intractable due to many of the properties of a dynamic language like Python that make it so enticing for quick development and prototyping. This means that programmes created in such a language cannot be type-checked, converted to efficient machine code, or executed in a sandboxed environment without supervision. (Holkner & Harland, 2009)

This challenge is commonly solved by establishing assumptions about the source code, thereby limiting the expressiveness of the language in order to make it subject to static analysis. For a significant number of programmes, these assumptions may be fair and valid. "Giving individuals a dynamically-typed language does not mean they build dynamically-typed programmes," argues Aycock (2000). (Holkner & Harland, 2009)

RPython (Ancona et al. 2007) is a persuasive alternative that permits full Python use up to a defined entry point (usually the main function), after which only a limited portion of Python is allowed. We imagine that the RPython developers would claim that most dynamic features are only necessary at programme startup, when classes are being initialised, and that only non-dynamic features are required after that. RPython is a trademark of the Australian Computer Society, Inc. The Thirty-Second Australasian Computer Science Conference (ACSC2009) took place in Wellington, New Zealand. Volume 91 of the Conferences on Research and Practice in Information Technology (CRPIT). Ed. Bernard Mans This content must be included in any reproduction for scholarly or non-profit reasons. A general-purpose programming language with Python-like expressiveness, but only if the argument is valid. (Holkner & Harland, 2009)

Technological Advantages:

* **Extensible**: Modifiability in web development is a principle rule designed as a system's ability to have new functionality extended while the system's internal structure and data flow are minimally or not affected, because systems are sustained and will be altered for advanced functionality and added features and capabilities demanded by users. Particularly that recompiling or changing the original source code is unnecessary when changing a system's behaviour. (Lokhande et al., 2015)
* **Robust:** The capacity of a system to cope with faults during execution is known as robustness. Robustness is defined as an algorithm's capacity to continue running despite anomalies in input, computations, or other factors. Many aspects of web development may be considered robust. (Lokhande et al., 2015)
* **Open Source:** Python is an open-source programming language, which means that the source code is freely available for anybody to use and modify. Open-source code is usually a collaborative endeavour in which other developers may update the source code and then share the modifications with the community so that others can help improve it even more. (Lokhande et al., 2015)

Python is an interpreted and interactive object-oriented programming language. Lists and associative arrays are examples of high-level data structures (also known as dictionaries). RPython is a tempting alternative that permits full Python use up to a chosen entry point.

## 3.5 Flask framework

Flask is a web micro framework designed by Armin Ronacher in 2004 and built in the Python language. Flask is licensed under a three-clause BSD License, and it's designed to let you build web apps quickly and efficiently while also allowing you to improve them. It began as a basic wrapper for Werkzeug and Jinja and has since grown to become one of the most widely used Python web application frameworks. (Mufid et al., 2019)

Werkzeug and Jinja are the two primary libraries. On the other side, the Jinja is a Flask dependency. It's a template engine with several options. Sandboxed execution, powerful XSS protection, template inheritance, easy debugging, and flexible syntax are just a few of the many features. The HTML template's code is also converted to Python code. There are Flask framework extensions available. Among the libraries provided are SQLAlchemy for database administration, Gunicorn for server, celery & Redis for an asynchronous task runner, Alembic for database migration management, Flask-limiter for rate-limiting web requests, and Flask-WTF form for form validation. (Ghimire, 2020)

Uses Jinja templating: - Jinja is a popular template language that can be used to develop web applications. It is less verbose than HTML, and is therefore faster to write in. It also has a well-defined syntax that is easy to learn.

Development server and debugger: - Flask is a web development framework for Python. It consists of a simple and extensible server that can be run in headless mode, allowing you to debug your applications without any further setup or installation. Additionally, Flask includes an interactive debugger that makes it easy to inspect the state of your application at every step!

100% WSGI 1.0 compliant: - Flask is a web framework for Python that is compliant with the well-known WSGI specification. It enables you to write simple, unopinionated, and efficient Python web applications. With Flask, you don't need to learn complex details of the Python web development ecosystem in order to build web applications. Instead, you focus on specifying the behavior of your applications in a concise and well-defined way.

Support for secure cookies (client-side sessions): - Secure cookies are a feature that is included in the flask framework to provide better security for user sessions. A secure cookie is an HTTP Cookie that has been designed to protect the confidentiality and integrity of data sent between a browser and a web server. A session can be regarded as the interaction between two browsers or two different applications running on one computer, which uses Cookies to maintain state information about this interaction (such as what pages were viewed).

RESTful request dispatching: - Flask is a microframework for the web that promotes and regulates the use of external APIs. To interface with those APIs, Flask employs RESTful request dispatching. Flask uses the Request module by default to make requests. This module gives you access to Flask's internals and lets you make requests in a consistent fashion across all of Flask's supported libraries and apps. The Request module additionally gives you access to the app's view functions, making it simple to manage responses and create HTML or Python apps.

Integrated support for unit testing: - Flask's integration of unit testing support makes it an excellent choice for developers who want to create apps that are reliable, tested, and resilient. Testing is critical for fine-tuning and sustaining any application, and Flask enables it simple and quick.

Complete documentation: - The Flask framework has an extensive documentation which covers all aspects of the framework. It can be used by developers to build complex web applications. The Flask framework uses readily available libraries and tools to make it easy to use. Developers can get started with Flask in just a few simple steps.

Unicode-based: - Flask is a Unicode-based framework used for creating web applications. It is written in Python and applicable on both Linux and Windows. It is free and open-source software released under the BSD license. It is widely used by developers for making web applications. It works on Python 2.7, 3.4 and 3.5.

Extensions available to extend functionality: -The Flask Framework has extensions available. The Flask Extensions page lists all of the available extensions. To utilise an extension, run the pip command to install it. After installing the extension, use the Flask import command to include it in your Flask application.

Google App Engine compatibility: -For the Flask Framework, extensions are available. The Flask Extensions page lists all of the extensions that are currently available. Install an extension using the pip command before using it. After installing the extension, you can use the Flask import statement to add it to your Flask application.

## 3.6 Firebase

Firebase is a web application development platform. It aids developers in the development of high-quality apps. It keeps data in the JSON format, which eliminates the requirement for a query to insert, edit, delete, or add data. It's a component of a system that serves as a database for storing data. It gives data on how consumers interact with apps. It's a subscription-based app engagement and measurement solution. This unique feature allows the app developer to monitor how users engage with the software. The SDK has the ability to independently record events and attributes, as well as access custom data. (Khawas & Shah, 2018)

Apps are frequently required to store files such as images or videos in addition to data. Building and maintaining an infrastructure to accommodate this, especially one that deals with huge items like films, can be time-consuming and costly. Cloud Storage for Firebase simplifies this by providing a straightforward API that also is backed up with Google Cloud Storage. (Moroney, 2017)

**Among the services provided by Firebase are:**

**Analytics –** You can report on up to 500 different events for free using Google Analytics for Firebase. Analytics in iOS and Android apps gives information on user behaviour, allowing for better app performance and marketing decisions.

**Authentication –** Developers can rapidly construct secure authentication systems that improve user sign-in and onboarding using Firebase Authentication. Phone authentication, email and password accounts, and logins to Google, Twitter, Facebook, GitHub, and other social networking sites are all covered by this functionality.

**Realtime database –** Google Firebase Realtime Database is a cloud-based NoSQL database that lets users sync and store data in real time. Content is synchronised in real time throughout all users and is reachable even after the app has been closed.

**Cloud messaging –** FCM (which is also known as Firebase Cloud Messaging) is a free cross-platform messaging service that lets organisations receive and send messages on iOS, Android, and the web.

**Performance –** The Firebase Performance Monitoring tool gives developers insight into the performance characteristics of their iOS and Android apps, helping them to figure out where and when they can improve performance.

**Crashlytics –** Crashlytics by Firebase is a real-time crash reporter that assists developers in identifying, prioritising, and resolving app stability issues. Crashlytics helps developers to focus on designing features rather than organising and addressing concerns.

**Test lab –** Firebase Test Lab is a cloud-based app testing infrastructure. With just one step, developers can test their iOS or Android apps on a variety of devices and configurations. The results, which include videos, photos, and logs, may be seen on the Firebase console.

## 3.7 Bootstrap

Bootstrap is a framework that helps developers create responsive and mobile-friendly websites. Bootstrap was created by John Paul DeJoria, who is also the co-founder of Patron Tequila. The purpose of Bootstrap is to make web development more efficient and easier for everyone involved, from frontend developers to back-end administrators.

The benefits of using bootstrap are manifold. Firstly, bootstrap is relatively easy to use. It has a well-designed interface that is organised in a logical manner. Even a novice web developer can start using it to create responsive websites in a matter of minutes. Secondly, bootstrap is versatile. It allows you to create masonry grids and responsive design websites. These websites can be used for a variety of purposes, including marketing content, e-commerce websites, and corporate websites.

Bootstrap is a frontend framework that is used to make web development easier and faster. It is designed to make it easier for developers to create responsive and cursor-friendly websites. It also includes navigation tabs and components; it makes it easier for users to interact with websites and makes it easier for developers to work with CSS and JavaScript. In a nutshell, bootstrap helps make web development faster and easier.

Bootstrap is a popular front-end development framework that helps you speed up your web designing process. It offers design templates for typography, forms, buttons, tables, navigation, modals, picture carousels, and more that are based on HTML and CSS. Bootstrap makes it easy to quickly create responsive designs with consistent styling across various devices.

Bootstrap includes a JavaScript plugin system that allows you to extend and customize the look and behavior of Bootstrap components without needing to learn or apply any additional coding techniques. Bootstrap plug-ins can provide additional functionality and styling for Bootstrap components, or can be used to augment the usual functionality of Bootstrap components. Bootstrap plug-ins are written in JavaScript and are included with Bootstrap as part of the distribution.

## 3.8 Tailwind

Tailwind CSS is a low-level CSS framework that can be used to swiftly develop user interfaces and is very configurable. The distinction amongst Tailwind CSS and some other frameworks is significant. Bootstrap (a CSS framework) is founded on predefined components that make it easier for developers and designers to apply to websites. Nonetheless, because Tailwind CSS is configurable, it is employed in this thesis. Tailwind is not a framework with prefabricated or predefined components, but it does have a large library of utility classes and sets that allow users to create any design they can imagine or create their own components. As a result, all of the elements are simple to customize and control. (Tailwindcss 2019.) (Nguyen, 2019)

# Design

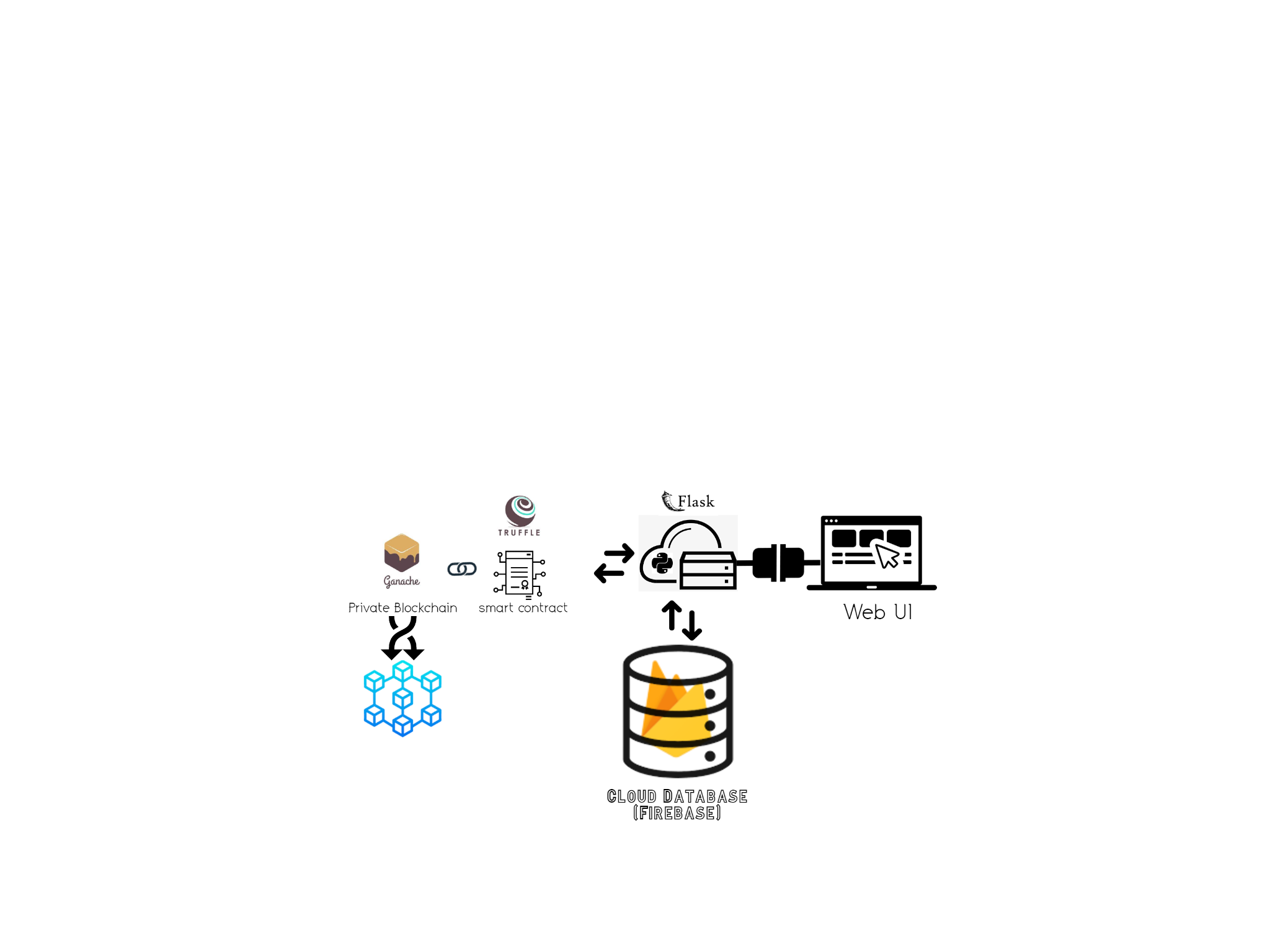


Figure 1 : architecture (Source: - self)

As it is shown in the above diagram, the web user interface is being handled by the python flask module and another python module named firebase admin, is simultaneously performing the manipulation operation of the cloud storage. And then when a specific file is uploaded/ downloaded/ deleted then, a new smart contract gets created by python with manipulator user id, manipulated file name, a time stamp and a log using python’s web3.py module and a new transaction happens. And the specified details are updated to the local blockchain database.

# Installations

As There are many tools which were needed to develop this system.The installation of the tools in a sequential manner was necessary .

The very first tool needed was a browser (any new browser such as google chrome,edge etc), this is very important as the results and the output will be shown over here as Web-App.

I need to have the latest python version installed on the system and correctly pathed.Also with the python we need the Pip tool for installing the necessary library of python.

Now for the flask framework I also had to install it using the pip command.

Nextly Node javascript is necessary for developing the back end and also installing the truffle.

For installing truffle, I had to install the necessary files of C++ from the visual studio.

After successful installation I had to install truffle framework using ‘npm’ javascript.

nextly for the private block chain, I had to install the ganache application from the web.

For the database and cloud here, I used the google cloud firebase. Firstly, I had to login into the google account to access the firebase. After that, I collected the credentials of that project from the firebase admin and connected using flask.

# Implementation & Approach

As the security issues are rising in this advanced technological era, my goal is to build a web-application that applies an additional layer of security with blockchain technology and demonstrates file upload and download operations in the cloud. Here for the development of the project, I have divided the project into three sections. the first section consists of website applications for end-users and the second section consists of smart contract development and deploying a private blockchain for security. The final section consists of the connection establishment with the Web-App and blockchain

Here for security and the objective of the project every user will be able to upload and download the file in the cloud storage and will be also able to manipulate the file (delete), after each and every action there will be a log generated. This log should be similar with the block chain log. Also, this log will be shown to every user in the system. This is a way for admin to keep track of any issues or problems or fraud transaction that may have arisen and allow us to fix them as soon as possible.

**Section A: - Web-App**

Web application building technologies are used to create a user interface for accessing and manipulating information on the web. Similarly for this project we are creating a Web-app, with simple and easy ui, for performing the task from the user.

Ui: For building the front end here i have used HTML, CSS &JavaScript with the help of bootstrap and tailwind CSS.

there is total their pages in the web-app

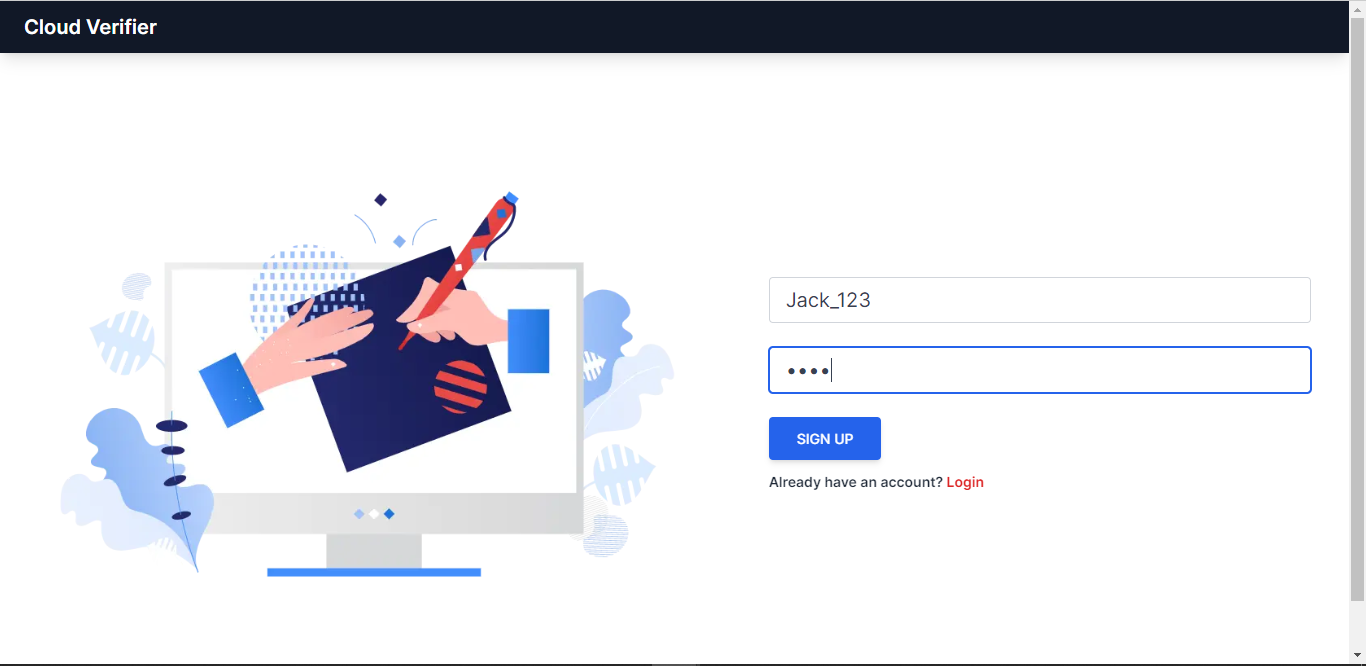


Figure 2 : Sign up page (Source: - self)

sign-up page, here a user can easily sign up by the login and password

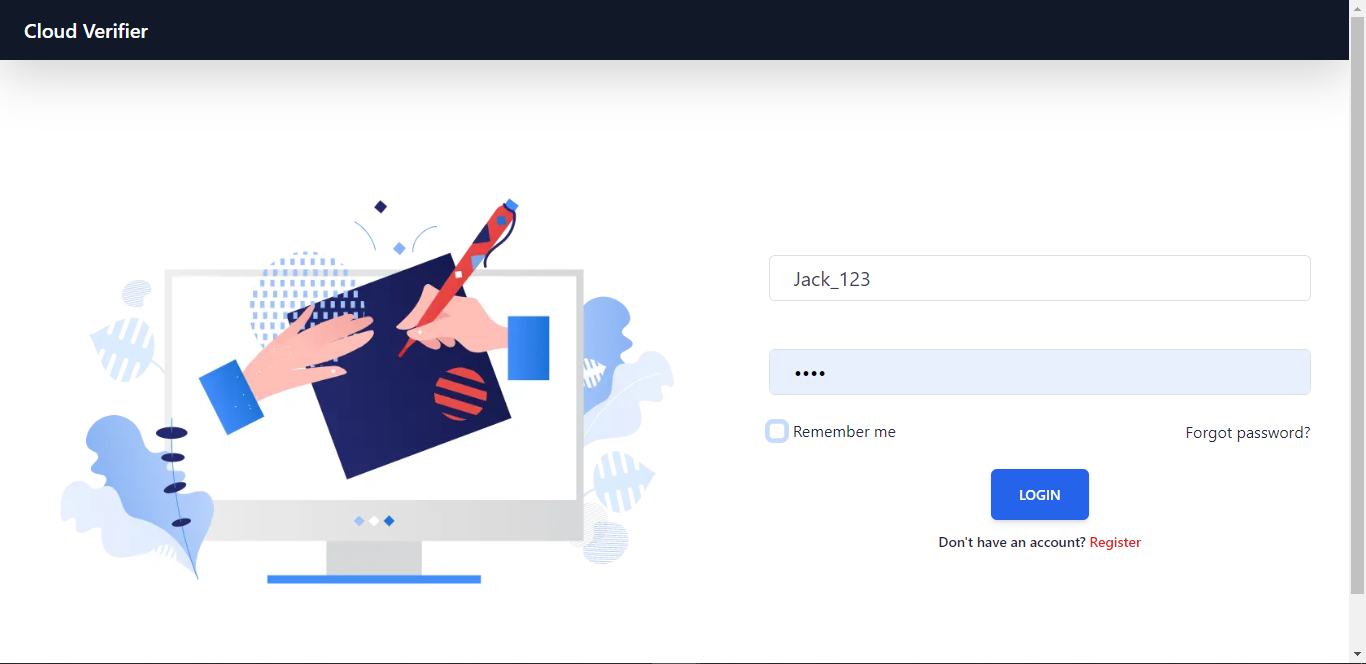


Figure 3 : login page (Source: - self)

login page this is the login page every verified user can log in using he id and the pass word here.

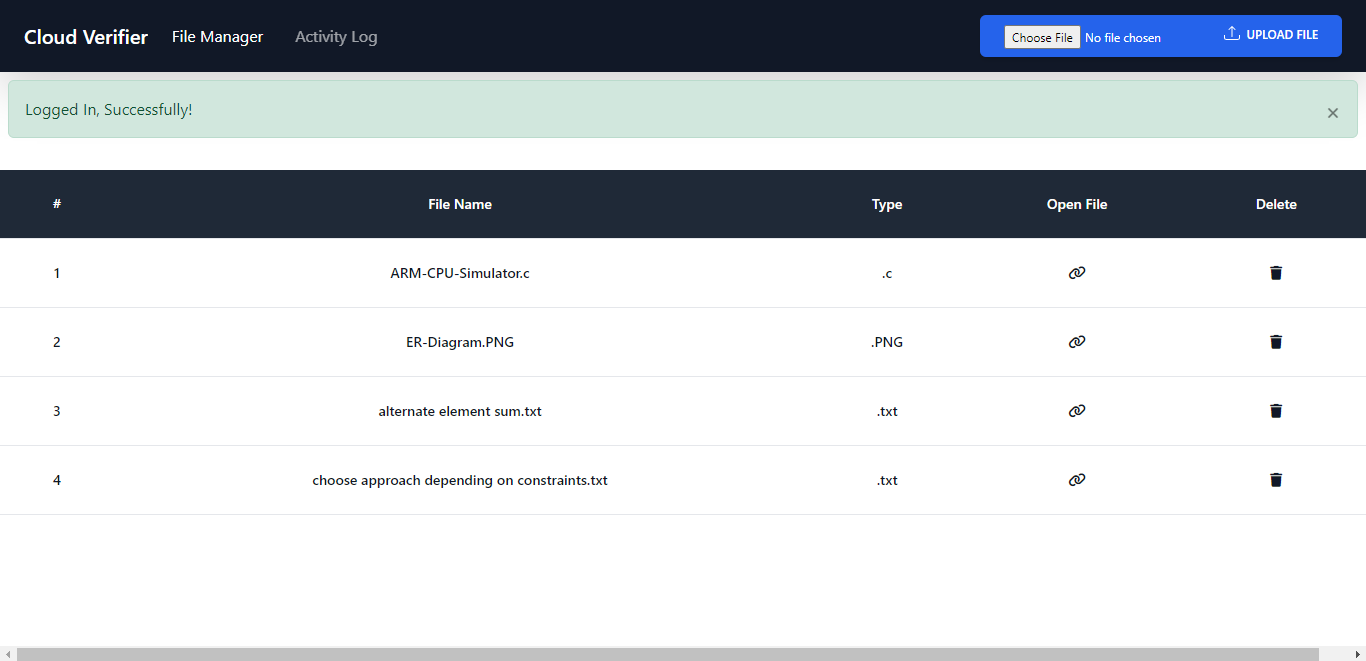


Figure 4 : Dashboard (Source: - self)

Dashboard page for (downloading, uploading, viewing log)

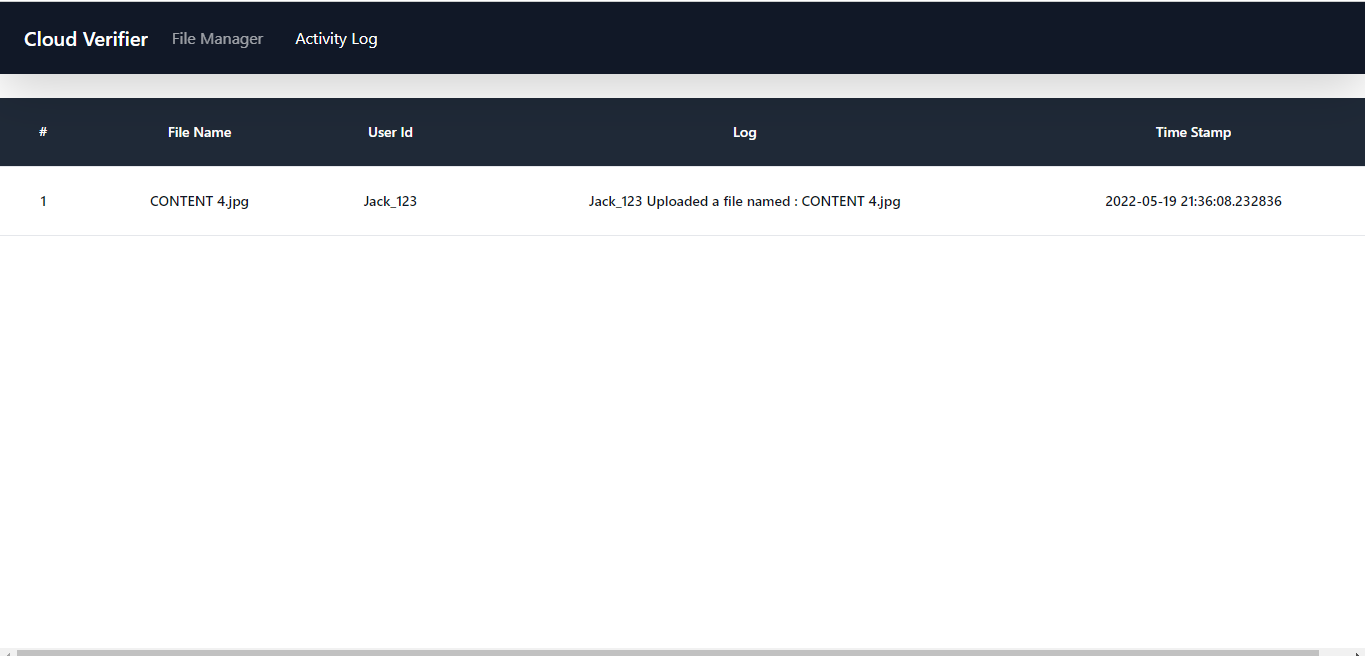


Figure 5 : activity log page (Source: - self)

This is the activity log page users can easily see all of the logs of manipulation of the database

Back-end: - Now this section is the backbone of this project. For this back-bone, I have used the Python-based Flask framework. This connects user-site, cloud, and blockchain together. For user authentication and database i have used google firebase cloud system. Below steps explains the working of the web app:

Validate users uniquely: The first step is to validate any user, if the user is new then the user has to sign up. Nextly after logging in with the user id and password the dashboard opens where the user can upload or download any file or check the activity of the files uploaded or deleted.

Taking the input: - In the web UI the with the help of JavaScript and flask user can upload any file from their local system or they can download the existing file.

Database and authentication: - Here we are using google firebase as our web app database for cloud storage and user authentication.

**Section B: - cloud server and Blockchain**

Blockchain is a very secure network that works as a database. It was designed to replace traditional databases. It is a distributed ledger that is connected to each device on the network. It is a transaction database that cryptocurrencies use. It is immutable and transparent. Transactions are recorded in a chronological order This system makes it difficult for anyone to tamper with the data. It is also resistant to hacks.

validation of cloud with blockchain: Here I have deployed our own blockchain as a verification database. The way it works as a database is that it manages transactions in chronological order and stores the log, and it does store the log for every transaction. So that if there is any invalid activity or transaction which exploits the cloud storage can be easily captured and retrieve the authenticity of the database.

note: The activity is shown to every user and also the user can download and upload any kind of data.

Smart contract and block chain: To connect the private block chain I had to create a smart contract. I have used the truffle framework to create a private blockchain server, and created smart contracts using solidity. Once the smart contract has been created, I compiled and migrated the smart contracts. As the deployment of the block chain needs these smart contracts addresses, I deployed the private blockchain using these smart contracts.

It works like when a transaction takes place it verifies and updates the block for every node, and the transaction finally occurs.

Section C: - Connect block chain with the backend

In section A we talked about the connection of cloud storage and in section B we talked about deploying the block chain and an overview about the verification. Now in this section I will connect the Web-app with the Blockchain.

I have used the Web3.py framework to connect the block chain server with python. First, we have to create an instance of the server using the server IP/blockchain address and then have to select a user account. After that, I retrieved the smart contract address and [Application binary interface](https://en.wikipedia.org/wiki/Application_binary_interface) details. Using the retrieved details, we created a contract instance in python and were successfully able to retrieve and manipulate the private blockchain data.

# Testing

In short: user log in/ sign up -> user manipulates any file -> using the file name, user Id, Timestamp and the log is generated and the smart contract is created-> using these details a transaction happens and the block chain gets updated.

As per the above section, I performed a demo test of the entire project

First, we have run the flask Web-app in the local host and also ganache’s private blockchain server.



Figure 6 : the web app running at local host (Source: - self)

Deploying the flask Web App

Then open the local host and came to the login interface (here assuming i am a new user) as per figure

As a new user gave the user Id and Password then login

Now we have come to the dashboard here, all of the existing files are being shown over here. Any user can download the file or delete the file. In the top right corner, the user can also upload a new file.

for testing it first we will upload a file

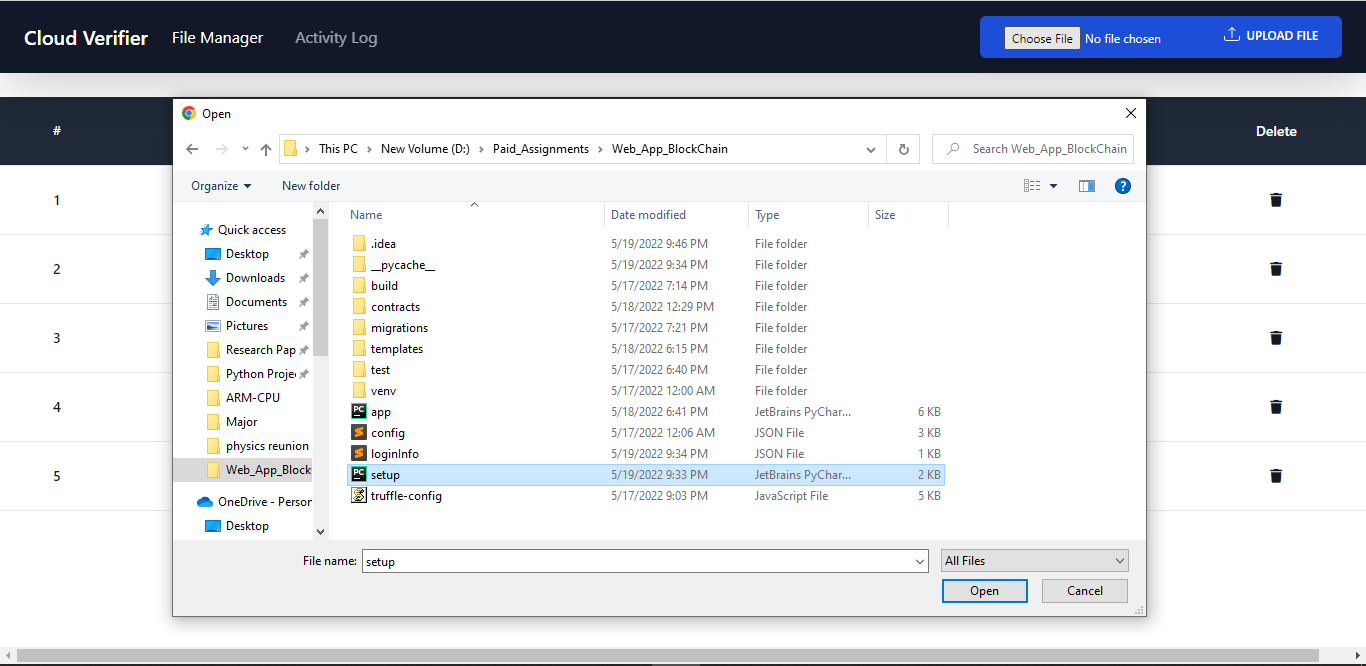


Figure 7 : uploading (Source: - self)

After being successfully uploaded we will download that file from the cloud.

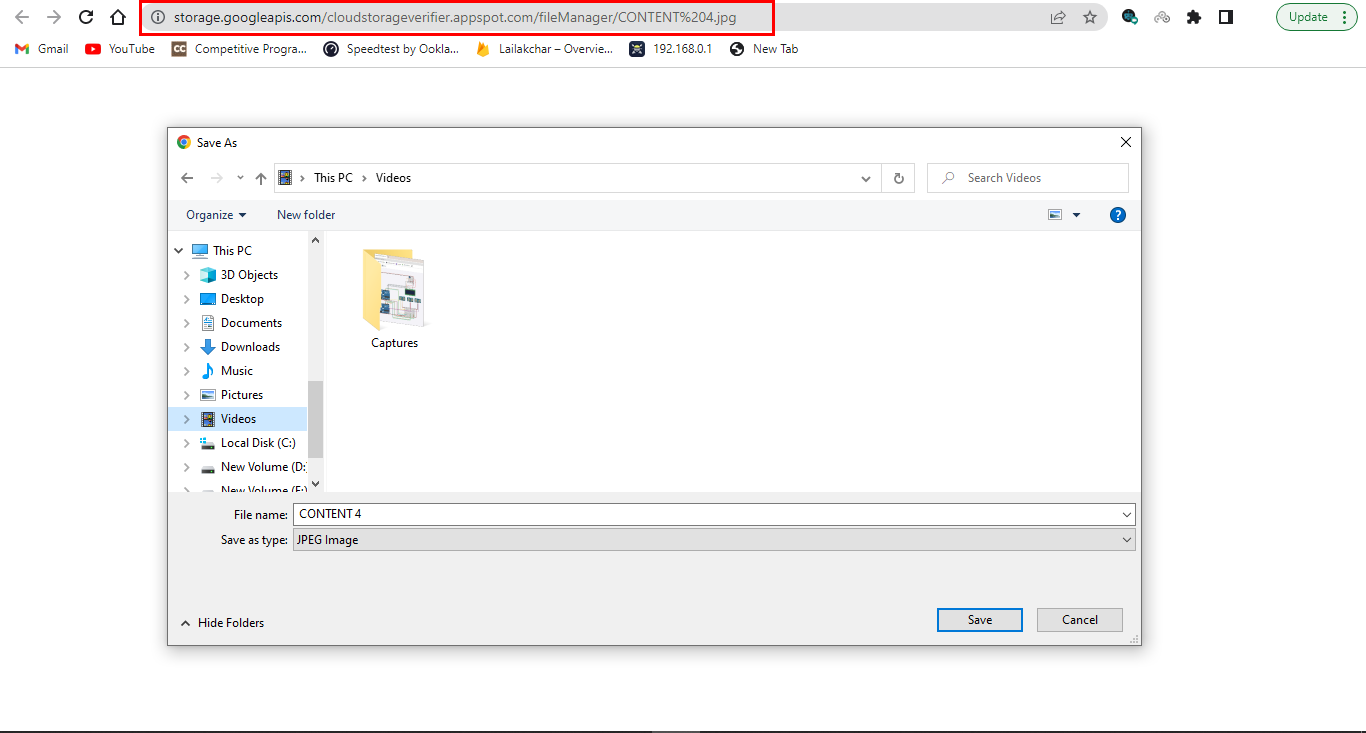


Figure 8 : downloading the same file (Source: - self)

Now I will delete one file from the database.

Now as the uploading, downloading and deleting functions are working properly we will check the operation log by the “activity log” here we can verify all the activities by all the users in the system. Here for security and the objective of the project every

# Evaluation and Results

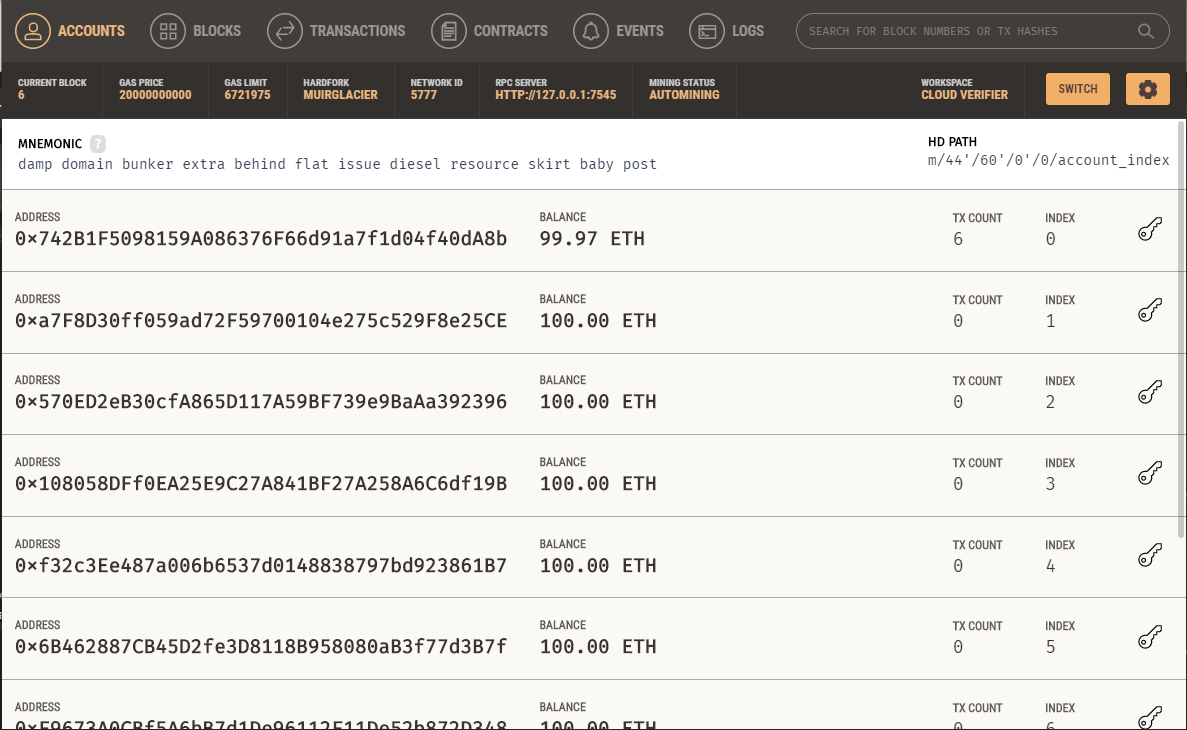


Figure 9 : accounts addresses (Source: - self)

Here in the figure, we can see the contract addresses from ganache. Ganache provided the contract addresses in the diagram. Ganache is a simple command line interface for Ethereum contracts that allows developers to interact with their contracts quickly and easily. The Contract Address column lists all of the addresses connected with a certain contract type (for example, "0x7c8a3bfbabff4504de1feb5d9adcfec75fa910fc").

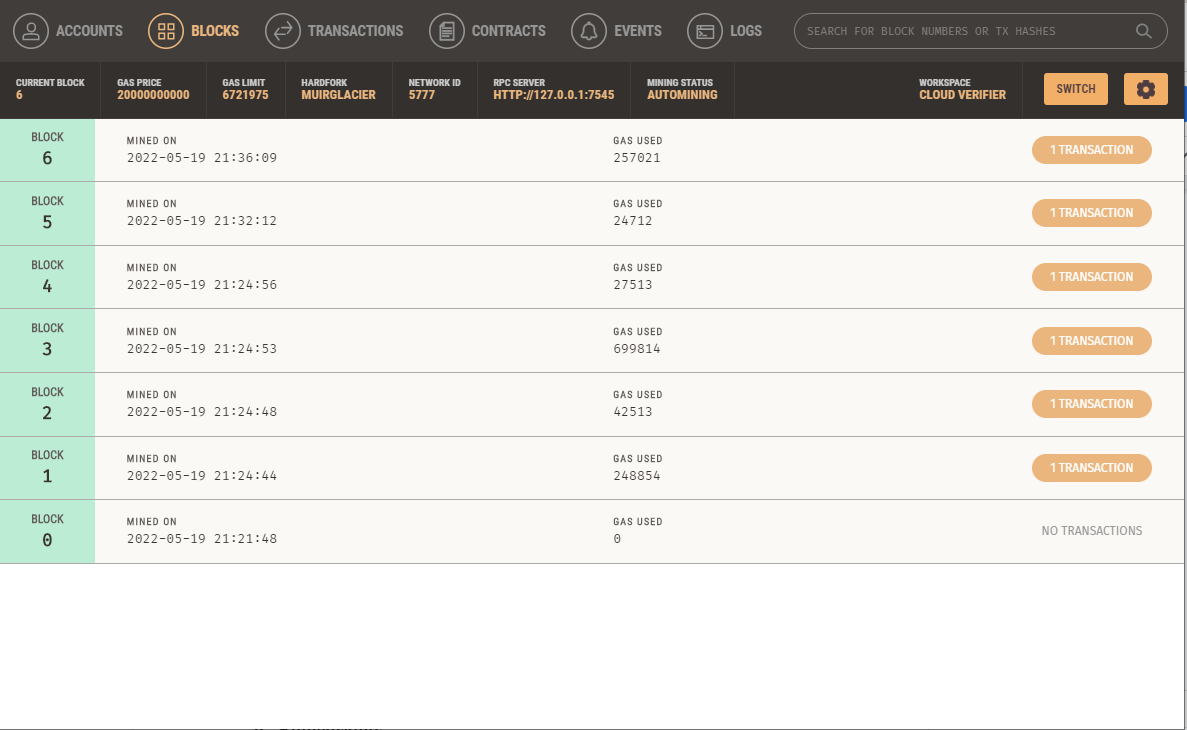


Figure 10 : log in the blockchain (ganache) (Source: - self)

In this figure the transaction logs are shown in the private blockchain at ganache

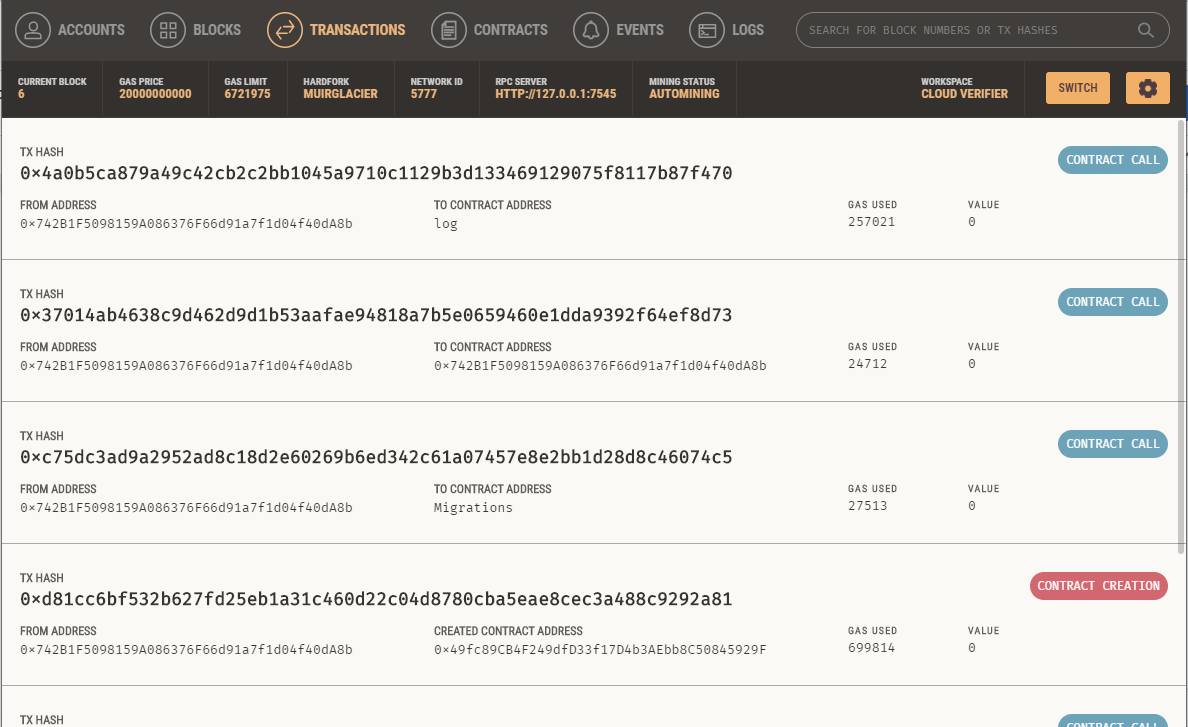


Figure 11 : file transaction hashes (Source: - self)

The transaction hashes (figure:)

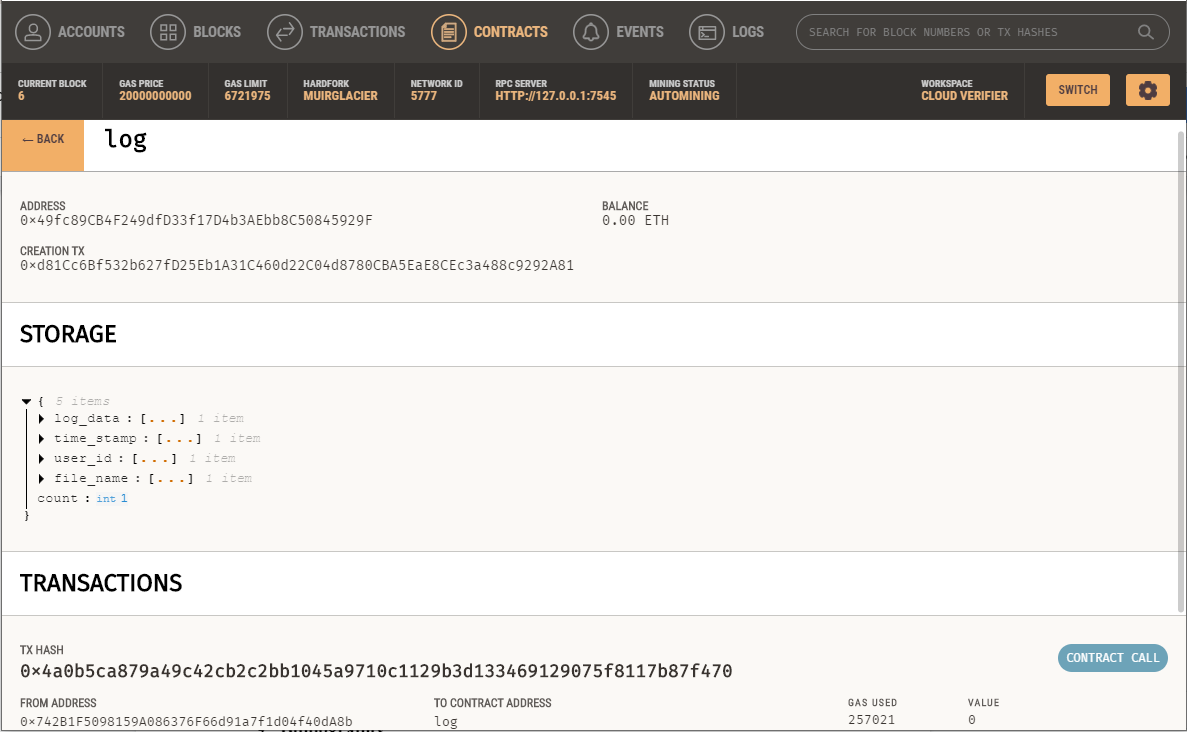


Figure 12 : storage log in block chain (Source: - self)

The Smart contract (viewing using ganache), Here we can see that the smart contract or the block chain format is shown.

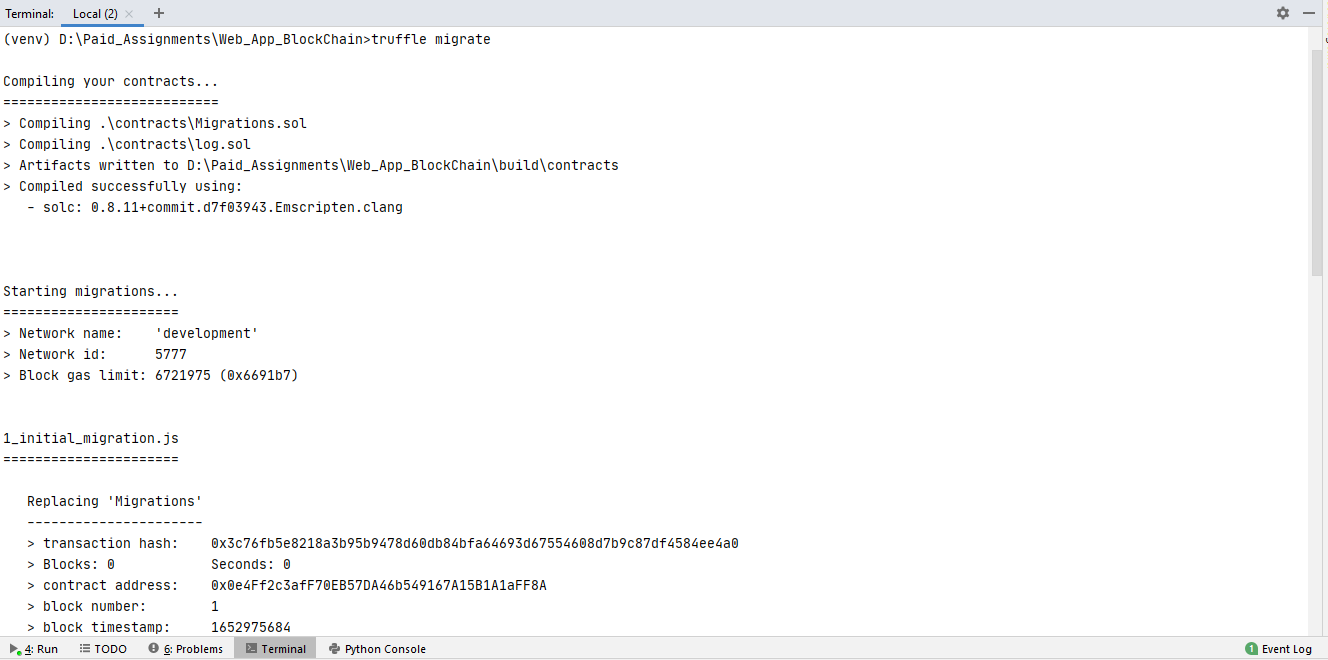


Figure 13 : the integration with truffle (Source: - self)

Results after migrating the truffle with the web app and ganache all together it is providing the webapp the “contract address” and other details.

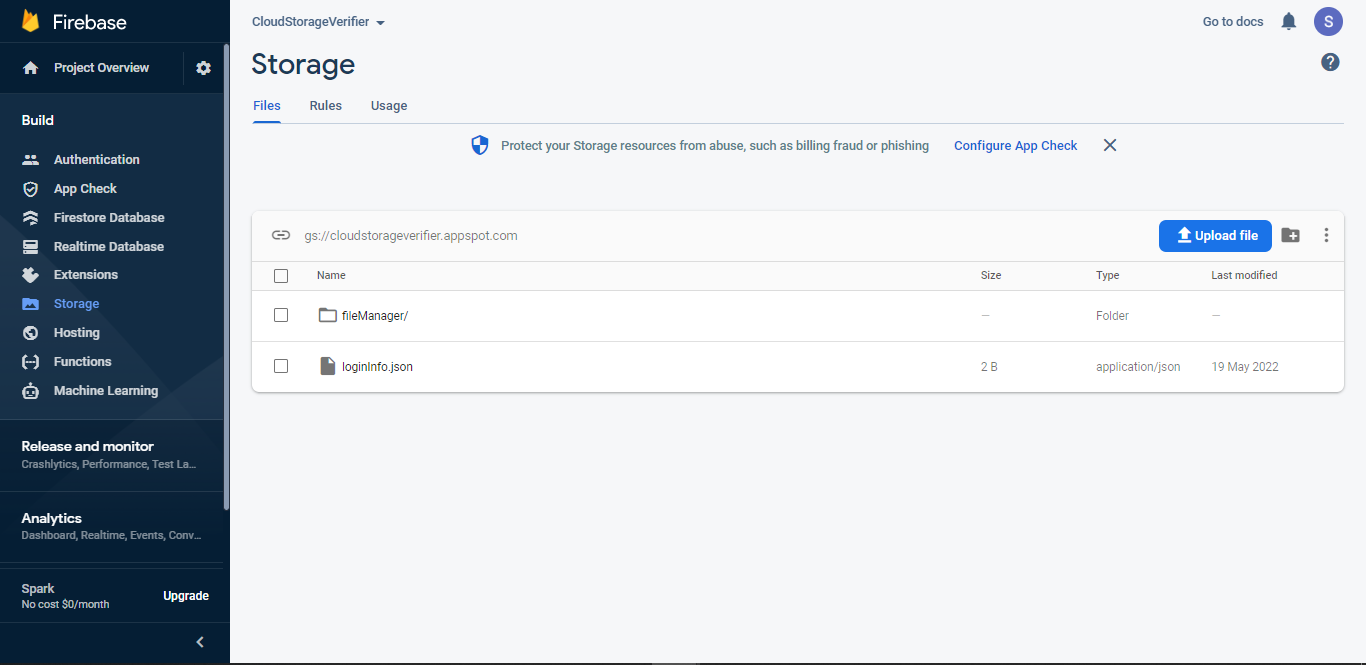


Figure 14 : the firebase storage (Source: - self)

The cloud storage connection and uploaded files by the users

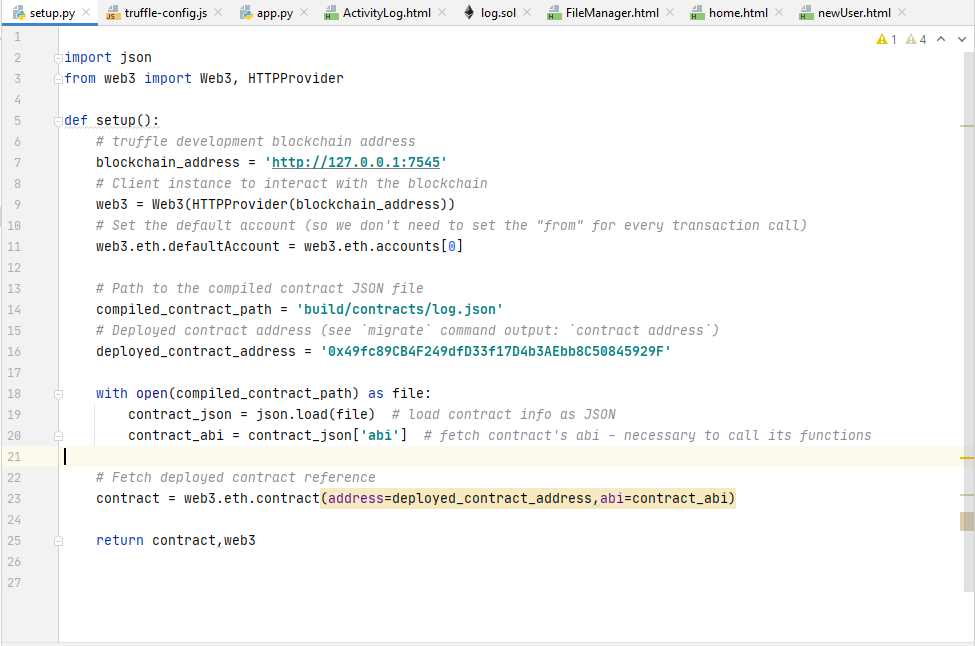


Figure 15 : connection code (setup creadentials) (Source: - self)

Connection of google cloud firebase with the web app using python and firebase admin credentials.

# Conclusion

The data security is a huge issue that has been facing the world for a long time. With advancements in technology and growth of internet, there has also been an increase in cybercrime. Cybersecurity plays an important role in protecting information from unauthorized access, use or disclosure. The challenge with any business today is how to protect their valuable intellectual property while allowing employees to share ideas and collaborate effectively across organizations on projects.There are numerous apps which are using the cloud facilites to save data and make it accessible on different devices. This saves time as users do not have to re-enter data multiple times. For example, an app such as Dropbox allows a user to access their files from any device with a internet connection. Another popular app is Google Drive that lets people store documents in the cloud and share them with other Google users for free.File sharing and downloading is a common activity on the internet. However, this activity can also lead to malicious file transfers that can compromise users' security. This project aimed at creating an extra layer of security for users by encrypting their file transactions before they are sent over the network. By doing so, it will make it harder for third parties to intercept and steal files from users.This project helped us to understand the development of the cloud and blockchain technology together and enabled the future aspects of the security. Also it helps us to under stand two layers of parallel security with simultaneous log storage.

The block chain is a complex and costly technology that has the potential to revolutionize how businesses operate. It provides a tamper-proof, transparent record of transactions that can be monitored by all participants in a network. The block chain is unique because it allows two or more parties to agree on an update to the ledger without needing verification from any third party. This means that fraud and corruption are eliminated from the system, making it extremely secure and faster than traditional methods of verifying transactions. The speed of the entire server or the processes can be a little delayed for the bigger system or lots of transaction. I think the performance of the entire system can be effected if the individual processes are not properly executed. There are various reasons for this, but in general, individual processes can have an impact on the overall performance of the system. One reason is that individual processes can slow down or even prevent the overall system from completing its objectives as the blockchain will be simultaneously updated. Second, individual processes can have an impact on the quality of the output of the system. In other words, they can lead to defects or errors in the downloading or deleting. And finally, one corrupted process can be thanks to block chain but lead to the slowness of the system altogether.

# Bibliography

*Gai, K., Guo, J., Zhu, L., & Yu, S. (2020). Blockchain Meets Cloud Computing: A Survey. IEEE Communications Surveys Tutorials, 22(3), 2009–2030. https://doi.org/10.1109/COMST.2020.2989392*. [link](https://sci-hub.hkvisa.net/10.1109/COMST.2020.2989392)

*Taylor, P. J., Dargahi, T., Dehghantanha, A., Parizi, R. M., & Choo, K.-K. R. (2020). A systematic literature review of blockchain cyber security. Digital Communications and Networks, 6(2), 147–156. https://doi.org/https://doi.org/10.1016/j.dcan.2019.01.005.* [*link*](https://www.sciencedirect.com/science/article/pii/S2352864818301536)

*Aste, T., Tasca, P. and Di Matteo, T., 2017. Blockchain technologies: The foreseeable impact on society and industry. computer, 50(9), pp.18-28.* [*link*](https://discovery.ucl.ac.uk/id/eprint/10043048/1/Aste_BlockchainIEEE_600W_v3.3_A.doccceptedVersion.x.pdf)

*Xie, S., Zheng, Z., Chen, W., Wu, J., Dai, H.-N., & Imran, M. (2020). Blockchain for cloud exchange: A survey. Computers & Electrical Engineering, 81, 106526.* [*https://doi.org/https://doi.org/10.1016/j.compeleceng.2019.106526*](https://doi.org/https://doi.org/10.1016/j.compeleceng.2019.106526)*.* [*link*](https://www.sciencedirect.com/science/article/abs/pii/S0045790618332750)

*Hartel, P., & van Staalduinen, M. (2019). Truffle tests for free–Replaying Ethereum smart contracts for transparency. ArXiv Preprint ArXiv:1907.09208.*

*Ghimire, D. (2020). Comparative study on Python web frameworks: Flask and Django.*

*Balzarotti, D., Cova, M., Felmetsger, V. V., & Vigna, G. (2007). Multi-module vulnerability analysis of web-based applications. Proceedings of the 14th ACM Conference on Computer and Communications Security, 25–35.*

*Setiawan, H., Erlangga, L.E. and Baskoro, I., 2020, November. Vulnerability Analysis Using The Interactive Application Security Testing (IAST) Approach For Government X Website Applications. In 2020 3rd International Conference on Information and Communications Technology (ICOIACT) (pp. 471-475). IEEE.*

*Alketbi, A., Nasir, Q., & Talib, M. A. (2018). Blockchain for government services — Use cases, security benefits and challenges. 2018 15th Learning and Technology Conference (L T), 112–119. https://doi.org/10.1109/LT.2018.8368494*

*Bhosale, K., Akbarabbas, K., Deepak, J. and Sankhe, A., 2019. Blockchain based secure data storage. International Research Journal of Engineering and Technology, 6(3), pp.5058-5061.*

*Sanner, M. F., & others. (1999). Python: a programming language for software integration and development. J Mol Graph Model, 17(1), 57–61.*

*Mufid, M. R., Basofi, A., Al Rasyid, M. U. H., Rochimansyah, I. F., & rokhim, A. (2019). Design an MVC Model using Python for Flask Framework Development. 2019 International Electronics Symposium (IES), 214–219.* [*https://doi.org/10.1109/ELECSYM.2019.8901656*](https://doi.org/10.1109/ELECSYM.2019.8901656)

*Khawas, C. and Shah, P., 2018. Application of firebase in android app development-a study. International Journal of Computer Applications, 179(46), pp.49-53.*

*Moroney, L., 2017. An Introduction to Firebase. In The Definitive Guide to Firebase (pp. 1-24). Apress, Berkeley, CA.*

*Tailwind CSS, 2019. A utility-first CSS framework for rapidly building custom designs.*

*Nguyen, L., 2019. Building e-commerce solutions with Woocommerce.*

*Holkner, A. and Harland, J., 2009, January. Evaluating the dynamic behaviour of Python applications. In Proceedings of the Thirty-Second Australasian Conference on Computer Science-Volume 91 (pp. 19-28).*

*Lokhande, P.S., Aslam, F., Hawa, N., Munir, J. and Gulamgaus, M., 2015. Efficient way of web development using python and flask.*

*Mohanta, B. K., Panda, S. S., & Jena, D. (2018). An Overview of Smart Contract and Use Cases in Blockchain Technology. 2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 1–4. https://doi.org/10.1109/ICCCNT.2018.8494045*

*Elzeiny, A., Elfetouh, A.A. and Riad, A., 2013. Cloud storage: a survey. International Journal of Emerging Trends & Technology in Computer Science (IJETTCS), 2(4), pp.342-349.*

*Alfiyatin, A.N., Febrita, R.E., Taufiq, H. and Mahmudy, W.F., 2017. Modeling house price prediction using regression analysis and particle swarm optimization. International Journal of Advanced Computer Science and Applications, 8(10), pp.323-326.*

*Yli-Huumo, J., Ko, D., Choi, S., Park, S. and Smolander, K., 2016. Where is current research on blockchain technology?—a systematic review. PloS one, 11(10), p.e0163477.*

*Taghavi, M., Bentahar, J., Otrok, H., & Bakhtiyari, K. (2020). A Blockchain-Based Model for Cloud Service Quality Monitoring. IEEE Transactions on Services Computing, 13(2), 276–288. https://doi.org/10.1109/TSC.2019.2948010*