الجمهورية الجزائرية الديمقراطية الشعبية

Republique Algérienne Démocratique Populaire
وزارة التعليم العالي والبحث العلمي

Ministère de l'Enseignement Supérieur de la Recherche Scientifique

Université 20 août 1955 Skikda — **جامعة 20 أوت 1955 – سكيكدة**

Faculte des Sciences
Departement d'informatique

كلية العلوم
قسم الإعلام الآلي

الرقم: ... / ق/ ا/ ا/ ا/ 2026

Autorisation de Dépôt de Mémoire de Master

Je soussigné : ..... Aliguechi Fawzia .....

Certifie que l'étudiant(e) : ..... Lechehab Naima .....

Spécialité : ..... I.A .....

Ayant soutenu le projet intitulé : A comparative study of Data storage in cloud Computing and Blockchain Technologies

A apporté les corrections nécessaires sur son manuscrit de Master.

Signature de l'encadreur

الجمهورية الجزائرية الديمقراطية الشعبية

Republique Algérienne Démocratique Populaire
وزارة التعليم العالي والبحث العلمي

Ministère de l'Enseignement Supérieur de la Recherche Scientifique



Université 20 août 1955 Skikda

Faculte des Sciences
Departement d'informatique

جامعة 20 أوت 1955 – سكيكدة

كلية العلوم
قسم الإعلام الآلي
الرقم: .... / ق/ إ/ أ/ ت.ع/ 2026

Autorisation de Dépôt de Mémoire de Master

Je soussigné : ..... Rebiguechri Fawzia .....

Certifie que l'étudiant(e) : ..... Kedadra Selma .....

Spécialité : ..... IA .....

Ayant soutenu le projet intitulé : A comparative study of Data storage in cloud Computing and Blockchain Technologies

A apporté les corrections nécessaires sur son manuscrit de Master.

Signature de l'encadreur



RÉPUBLIQUE ALGÉRIENNE PULAIRE
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITÉ DU 20 AOÛT 1955 SKIKD DÉMOCRATIQUE ET POA

Faculty of Science
Department of Computer Science
Master's Thesis in Artificial Intelligence



GRADUATION THESIS

Topic:

A Comparatif Study Of Data Storage In Cloud Computing And Blockchain Technologie

Presented by:
-KEDADRA Selma
-LECHEHEB Naima

Supervised by:
Dr.ALI GUECHI Farida

Academic Year: 2025/2026

Acknowledgments

Acknowledgments

We proudly dedicate this research work:

To our beloved parents, the source of our strength, motivation, and success. Thank you for your endless love, unconditional support, and prayers that guided us through every step of this journey.

To our families and siblings, for their constant encouragement and for always believing in our potential.

To our respected supervisor, for her invaluable guidance, patience, and academic support throughout the development of this project.

To each other, as partners, for the exceptional teamwork, shared late nights, and mutual support that turned every challenge into a success. Finally, to all our friends and everyone who, even with a kind word, contributed to making this dream a reality.

Abstract

Abstract

Cloud Computing and Blockchain are two key technologies in modern information systems. While Cloud Computing provides scalable and flexible resources for data storage and processing, Blockchain offers a decentralized and secure environment that ensures data integrity and transparency. This dissertation presents a comparative study between the two technologies with a focus on security aspects. A practical application was developed using MongoDB Atlas and BigchainDB to evaluate their performance and security. The results show that Cloud Computing offers greater flexibility and efficiency, whereas Blockchain provides enhanced mechanisms for ensuring data integrity, traceability, and transparency. The study highlights the strengths and limitations of both technologies and demonstrates their potential contribution to building secure and reliable information systems.

Keywords: Blockchain, Cloud Computing, Security, Data Integrity, Privacy, Information Systems.

Table of contents

Table of contents

Acknowledgments II
Abstract II
Table of contents III
Table of figures and Table III
General Introduction 1
Chapiter 01: Blockchain 3
Introduction: 3
1. Historical Background of Blockchain 3
1. Définition 4
2. TYPES OF BLOCKCHAIN 5
2.1. Public Blockchain 5
2.2. Privateblockchain 5
2.3. Consortium blockchain 5
3. Blockchain Generations 6
3.1. Blockchain 1.0: Cryptocurrencies 6
3.2. Blockchain 2.0: Smart Contracts 7
3.3. .Blockchain 3.0: DApps 8
3.4. Blockchain 4.0: Use in Industry 8
4. Blockchain Platforms 9
4.1. Bitcoin 9
4.2. Hyperledger 9
4.3. Litecoin 10
4.4. Ethereum 11
4.5. Ripple 12
Figure1. 6: logo Ripple 12

5. Blockchain structure 13
5.1. The Chain of Blocks 13
5.2. Block 13
5.3. Block Header 14
5.4. Block Body 14
5.5. Consensus algorithm 14
6. Advantages of Blockchain 15
7. Blockchain Challenges 16
8. How Blockchain Works? 18
9. Blockchain Security 18
10. Security aspects in Blockchain networks 19
11. PRIVACY AND SECURITY TECHNIQUES USED IN BLOCKCHAIN 20
11.1. Anonymous Signatures 20
11.2. The Trusted Execution Environment (TEE) Based Smart Contracts 21
12. Blockchain as a security solution 24
12.1. Data usage security 24
12.2. System Security 25
CONCLUSION 25
Chapitre02: Cloud computing 26
Introduction 26
1. Cloud Computing 26
1.1. History 26
1.2. Définition du Cloud Computing 28
1.3. Characteristics of cloud Computing 28
1.4. Cloud Computing Architecture 29
1.5. Cloud Models 29
1.6. Cloud Platform 34
1.7. Cloud Providers 35
1.8. Advantages of Cloud Comiting 36
2. Cloud Security 38
2.1. Need for Data Security in Clouds 38
2.2. Security issues in the cloud 39
2.3. CLOUD COMPUTING SECURITY CONCERNS 40
2.4. Fundamentals of security 41

2.5. Security mechanisms 42
3. Security Comparison Between Cloud and Blockchain.....46
3.1. Security Dimension: Architecture & Control.....46
3.2. Security Dimension: Data Integrity 47
3.3. Security Dimension: System Availability 47
3.4. Security Dimension: Trust Model.....47
3.5. Security Dimension: Access Control & Encryption.....47
4. Comparative Analysis of Cloud and Blockchain Security 48
5. INTEGRATION OF CLOUD AND BLOCKCHAIN 48
5.1. BLOCKCHAIN SUPPORT FOR CLOUD COMPUTING.....48
Conclusion.....50
Chapitre03: Application Design and UML Modeling.....51
Introduction 51
1. What is UML?.....51
2. Unified Modeling Language (UML) Diagrams 52
3. SYSTEM REQUIREMENTS ANALYSIS 52
3.1. Identification of Acteurs 53
3.2. Functional Requirements.....53
3.3. Non-Functional Requirements.....54
3. System Diagrams Documentation.....55
3.1. Use Case Diagram 55
3.2. Activity Diagrams 56
3.3. Sequence Diagrams.....63
Conclusion.....68
Chapitre04:System Implementation and Performance Evaluation 69
Introduction 69
1. utils de développement et langages utilisés.....69
1.1. Eclipse 70
1.2. Java Environment.....70
1.3. MongoDB Atlas 71
1.4. BigchainDB 71
2. System Implementation and Performance Evaluation 72
2.1. User Interfaces Description.....72
Conclusion.....82

General Conclusion 83

List of References 85

Table of figures and Table

Table of figures and Table

Table1. 1 *: Comparison of Blockchain Types[5]* 6
Table1. 2 *: Comparative Analysis of Cloud and Blockchain Security[47].* 48

Table3. 1 *: Functional Requirements* 54

Table4.1 *: Data Ingestion Performance (Storage Tests)* 79
Table4. 2 *Data Retrieval and Display Performance (Query Tests)* 80

Figure1. 1 *: Structure of a Decentralized Blockchain Network* 4
Figure1. 2 *: Logo Bitcoin* 9
Figure1. 3 *; Logo Hyperledger* 9
Figure1. 4 *: Logo Litecoin* 10
Figure1. 5 *: Logo Ethereum* 11
Figure1. 6 *: logo Ripple* 12
Figure1. 7 *: Sequential Block Linking Mechanism[13].* 13
Figure1. 8 *: Block Structure[14]* 15
Figure1. 9 *:Blockchain worksStructure[18].* 18
Figure1. 10 *: Symmetric Cryptography[24].* 23
Figure1. 11 *: Asymmetric Cryptography[24].* 23

Figure2. 1:Cloud Cloud Models. 30
Figure2. 2: LaaS Infrastructure Model[32]. 31
Figure2. 3: PaaS Infrastructure Model[32] 32
Figure2. 4: SaaS Infrastructure Model [32]. 33
Figure2. 5: Architecture modèle de maturité Zero Trust [45]. 45
Figure2. 6:Architecture Zero Trust Lincoln Laboratory[45]. 46
Figure2. 7: Benefits that Blockchain providing to cloud [48]. 50

Figure2. 1 *:Cloud Cloud Models.* 30
Figure2. 2 *: LaaS Infrastructure Model[32].* 31
Figure2. 3 : *PaaS Infrastructure Model[32]* 32
Figure2. 4 *: SaaS Infrastructure Model [32].* 33

Figure2. 5 : *Architecture modèle de maturité Zero Trust [45].*.....45
Figure2. 6 :*Architecture Zero Trust Lincoln Laboratory[45].* 46
Figure2. 7 : *Benefits that Blockchain providing to cloud [48].* 50

Figure4. 1: logo Eclipse *70*
Figure4. 2: logo java *70*
Figure4 3 : MongoDB Atlas *71*
Figure4. 4: BigchainDB *71*
Figure4. 5 : Main Application Dashboard (HomeForm) *72*
Figure4. 6: Individual Student Registration Interface.....*73*
Figure4. 7: Automated Bulk Registration and Benchmark Interface (BulkRegistrationForm) *74*
Figure4. 8 : Synchronized Cloud Storage Data Viewer (DisplayCloudyDataForm).....*75*
Figure4. 9 : Cloud Storage Query Panel (CloudSearchForm) *76*
Figure4. 10 : Decentralized Ledger Data Viewer *77*
Figure4. 11 : Ledger Search and Verification Interface *78*
Figure4. 12: Data Ingestion Performance Curve (Storage).....*79*
Figure4. 13 : Data Ingestion Performance curve Curve (Storage).....*80*

General Introduction

General Introduction

Digital transformation has significantly changed the way information is stored, processed, and exchanged. As organizations increasingly rely on digital technologies, ensuring data security, transparency, and efficient resource management has become a major challenge. Among the emerging technologies that have gained considerable attention is Blockchain. Initially introduced as the underlying technology of cryptocurrencies, Blockchain has evolved into a powerful decentralized system capable of ensuring data integrity, transparency, traceability, and security without the need for a central authority. Its unique characteristics have made it applicable in various sectors such as finance, healthcare, supply chain management, and cybersecurity.

At the same time, Cloud Computing has become one of the most widely adopted computing paradigms. It provides on-demand access to computing resources, storage, and services through the Internet, offering flexibility, scalability, and cost efficiency. However, despite its numerous advantages, cloud environments continue to face several security and privacy challenges. Given the complementary strengths of these two technologies, integrating Blockchain with Cloud Computing has emerged as a promising approach to improve trust, data protection, and security in cloud environments.

This dissertation aims to study Blockchain technology and Cloud Computing, explore their architectures, characteristics, advantages, and security aspects, and analyze how Blockchain can contribute to addressing the security challenges encountered in cloud-based systems Dissertation Structure To achieve the objectives

of this study, this dissertation is structured into four main chapters, organized as follows:

Chapter 1: Presents the fundamental concepts of Blockchain technology, including its architecture, types, platforms, and security mechanisms.

Chapter 2: Introduces Cloud Computing concepts, security challenges, protection frameworks, the comparative betweenblocchaine vs cloudcomputing and the integration of cloud and blockchain technologies.

Chapter 3: Focuses on the design and modeling of the proposed system using UML diagrams and defines its overall architecture.

Chapter 4: Describes the implementation of the application, the development tools used, and the presentation of the final system interfaces and functionalities;

Blockchain

Chapiter 01

Chapiter 01: Blockchain

Introduction:

Blockchain is one of the most important technical invention in the recent years. Blockchain is a transparent money exchange system that has transformed the way a business is conducted. Companies and tech giants have started investing significantly in the blockchain market and it is expected to be net worth of more than 3 trillion dollars in next 5 years. It has become growing popular because of its irrefutable security and ability to provide complete solution to digital identity issues. It is a digital ledger in a peer to peer network. This chapitre provides a background on Blockchain technology, history, it's architecture, how it works, advantages and disadvantages and its application in different industries.

1. Historical Background of Blockchain

Blockchain technology did not appear suddenly, but it was the result of the development of the financial technology, and virtual currencies that occurred over decades. In fact, blockchain is the original technology that underlies cryptocurrency, the most famous of which is "Bitcoin", and the one for which blockchain technology was invented. As it was developed as the underlying infrastructure and database of recording and executing transactions for the digital currency bitcoin1 . The concept of blockchain was first fully conceived as an enabling technology for the bitcoin cryptosystem, as introduced in 2008 by a pseudonymous individual or group known as "Satoshi Nakamoto" in a research paper titled "Bitcoin: A Peer-to-Peer Electronic

Cash System”2 . Where he provides a peer-to-peer electronic cash system based on proof rather than trust party, allowing any two parties to transact and interact directly with each other without any third trust party such as bank. In this regard, cryptographic proof stands for the encrypted process of reaching validation, or what they have called consensus, through a proof of work system between the participants. That eliminates the need of any intermediary in which they trusted in3 . Moreover, this technology comes as a solution of the double spending problem associated with digital currency that arises when one party sends the same digital currency (token) to several different parties, creating new tokens, one for each different party, and essentially spending the same digital cash more than once[1].

1. Définition

la blockchain se définit comme “A Blockchain is a digital, immutable, distributed ledger that chronologically records transactions in a real time. The condition for each subsequent transaction to be added to the ledger is the consensus of the network participants (called nodes), thereby creating a continuous mechanism of monitoring regarding manipulation, faults, and data quality”[1].

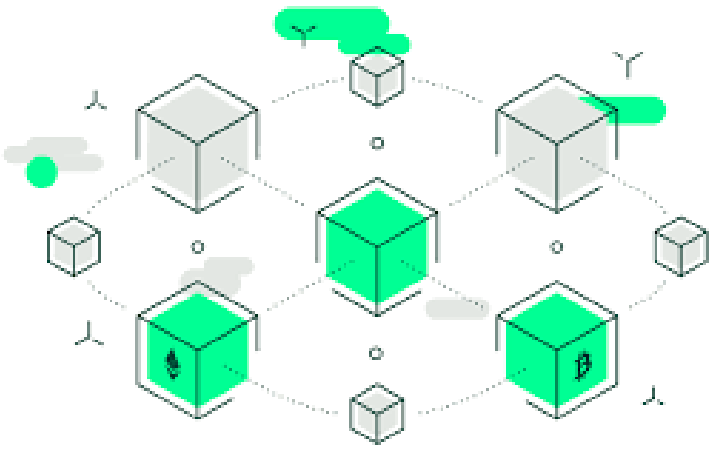


Figure1. 1: Structure of a Decentralized Blockchain Network

2. TYPES OF BLOCKCHAIN

2.1. Public Blockchain

It is a public infrastructure network, i.e. anyone is free to join the network without permission. Network participants can view and even participate, in and validate transactions. The most commonly used infrastructures are Ethereum and Bitcoin. Public BCs are mostly those applications open to the general public, where full transparency and decentralization are required. In addition, they encourage network members, called ‘‘miners’’; to check transactions to receive in return some rewards in the form of cryptocurrency [2].

2.2. Privateblockchain

The private blockchain is a limited one, where no one can quickly become a part of the network. It is a type of centralized blockchain controlled by a central authority for accessibility. The data read authorization in the private blockchain is opened to the public selectively. Private Blockchain is specific to limited organizations or small industries. Vote counting, Digital Identity, Asset Ownership, Supply chain management are different kinds of use cases of the private blockchain[3].

2.3. Consortium blockchain

A Consortium Blockchain is a hybrid of Private and Public Blockchains, where a group of companies or representatives come together to make decisions for the network’s benefit, rather than relying on one central authority. This approach speeds up decisionmaking and mitigates single points of failure. Consortium Blockchains offer options for rights and access management while leveraging the benefits of BCT. Examples of consortium Blockchains include R3 and EWF [4].

Feature	Public Blockchain	Consortium Blockchain	PrivateBlockchain
Authority	None	Multiple entities	Single defined entity
Access	Public, read and write	With authorization	With authorization
Identity	Anonymous	Accredited participants, known identities	Accredited participants, known identities
Security	Proof-of-Work / Proof-of-Stake	Predefined administrator	Predefined administrator
Consensus Participation	All nodes	Selected nodes	Defined entity
Speed	Slow	Fast	Fast

Table1. 1: Comparison of Blockchain Types[5]

3. Blockchain Generations

3.1. Blockchain 1.0: Cryptocurrencies

The first generation was entirely focused on decentralizing currency and payments, although it marked the first implementation of distributed ledger technology (DLT). It

underpins the operation of Bitcoin. The network is peer-to-peer, and transactions take place directly between users without the involvement of a third party. Other cryptocurrencies emerged, such as Litecoin, Dogecoin, etc. Bitcoin technology is based on the blockchain and a protocol used to describe how bitcoins are transferred. The consensus algorithm used is Proof of Work (PoW). Blockchain 1.0 ensures distributed storage, enables data sharing between nodes, and guarantees transaction transparency[6].

3.2. Blockchain 2.0: Smart Contracts

In blockchain “smart contract” is like a way of ensuring the two parties in interaction and to build a trust between them. Smart contract is a computer program which runs over the blockchain under the certain rules with defined by two or more parties as contract. This type of contract with enables the parties to interact with each other without any hesitation. As the defined conditions of the parties met, smart contract agreement automatically applied. Additionally, the smart contract code verifies the conditions, assists the parties and ensures the enforcement onnegotiation as per mentioned rules and regulation of contract of the parties. On-board parties need to submit their assets into the computer code based smart contract, all assets automatically divided in parties as per mentioned in the contract distribution agreement. These smart contracts are highly secure and impossible to tamper them. The smart contracts help out the parties to check and track performance in real time which is highly cost saving. In the traditional contract a third party involves as a guarantor, that is the main difference between smart contract and traditional contract. Smart contract have a capability to self-verify, self-execution and avoid from tampering. Vender machine is the real time best example of smart contracts. We purchase product from vender machine, we need to put money in vender machine and machine decide that either entered money is enough for the product or not. A program which installed in the machine will decide if money is enough, it will eject the product if not enough it will never eject the product. The machine will show other possible

options for user after analyzing the money. Vender machine gives 24/7 service to customers[7]

3.3. .Blockchain 3.0: DApps

Blockchain 3.0 focuses on justice, efficiency, and coordination applications beyond sectors uch as cryptocurrency, economics, and financial markets. Blockchain 3.0 is a new paradigm for organizing transactions and various activities with less friction and more efficiency. Novel applications of blockchain technology can be used to facilitate the coordination between human interactions and enable a higher order of collaboration for human-machine interaction. A blockchain solution is able to support both qualitative and quantitative data analysis and offer extrinsic and intrinsic benefits to stakeholders involved in the blockchain network. Blockchain 3.0 depicts the era of a programmable society. Blockchain 3.0 offers the potential to integrate the internet of things, artificial intelligence, and big data to provide highquality services for society. It is expected that blockchain technology will be more widely applied in the next few years across many industries and many aspects of people's daily life and work[8].

3.4. Blockchain 4.0: Use in Industry

Blockchain 4.0 represents the latest evolutionary stage of distributed ledger technology, specifically designed to meet the rigorous demands of Industry 4.0. this generation transcends simple transaction processing to provide a decentralized, secure, and autonomous infrastructure for CyberPhysical Systems. By integrating Artificial Intelligence (AI) and the Internet of Things (IoT), Blockchain 4.0 enables real-time, end-to-end management of industrial supply chains, fostering maximum transparency and trust without the need for specialized programming intermediarie [7].

4. Blockchain Platforms

4.1. Bitcoin



Figure1. 2: Logo Bitcoin

It is the first practical blockchain platform; it was presented in 2008 by a pseudonymous person known as "Satoshi Nakamoto". It is a public permissionless blockchain that uses peer-to-peer platform to conduct financial transactions and transferring money based on a cryptocurrency (bitcoin) without a central authority or banks. It is the first blockchain platform to be widely used, as its success is largely attributable to the creative use of the peerto-peer network to implement every stage of the currency's life cycle, from creation to transfer. Whereas enabling global transactions to be completed at the same pace as local transactions and providing a transaction history[1].

4.2. Hyperledger



Figure1. 3; Logo Hyperledger

Hyperledger is a global open-source project hosted by the Linux Foundation, created to promote cross-industry blockchain technologies. At the time of writing, Hyperledger comprises fourteen projects, six of which are distributed ledgers and the other eight are supporting modules. This project, called Fabric, features a highly modular and configurable architecture, promoting innovation, versatility, and optimization for a wide range of use cases across various sectors, including banking, finance, insurance, healthcare, human resources, and supply chain. Fabric is the first distributed ledger platform to support smart contracts (called Chaincode) written in general-purpose programming languages such as Java, Go, and Node.js, and to run in containers (Docker) [2].

4.3. Litecoin



Figure1. 4: Logo Litecoin

Litecoin is a peer-to-peer digital currency that enables instant, near-zero-cost payments to anyone in the world. It is an open-source, fully decentralized global payment network with no central authority, secured by mathematics to enable individual control over finances. It offers faster transaction confirmation times and greater storage efficiency than Bitcoin, with strong industry support, trading volume, and liquidity, complementing Bitcoin. The Litecoin blockchain handles a higher transaction volume thanks to more frequent block generation, currently rewarding miners with 6.25 LTC per block (halved approximately every 4 years), for a total of 84 million units[9].

4.4. Ethereum



Figure1. 5: Logo Ethereum

Ethereum is a decentralized blockchain network and an open-source software development platform that uses the cryptocurrency Ether (ETH) as the primary fuel for transactions. Launched in July 2015 by developer Vitalik Buterin and a small team of co-founders, it represents an evolution of the Bitcoin concept by integrating smart contracts end open-source programs that enable the creation of digital assets and decentralized applications (dApps) operating 24/7 on a global scale, without the need for banks or intermediaries. Today, Ethereum hosts thousands of cryptocurrencies and applications in areas such as decentralized finance (DeFi), non-fungible tokens (NFTs), gaming, decentralized social networks, and stablecoins, forming a thriving ecosystem known as Web3, the third phase of the internet focused on ownership and decentralization[10].

4.5. Ripple

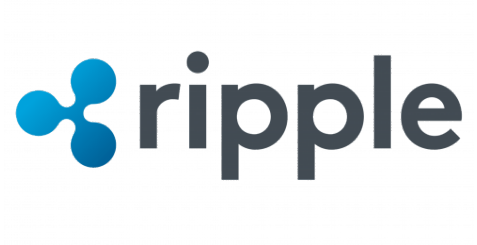


Figure1. 6: logo Ripple

It is an open source protocol created to facilitate quick and inexpensive financial transactions. It employs a shared ledger that is controlled by an interconnected system of autonomous nodes. It is regarded as one of the most cutting-edge platforms for distributed ledger technology (DLT), as only focuses on using blockchain for payments and remittancesector. It has been implemented by a growing number of financial organizations to offer alternative transfer options to users. Even more, the New York State Department of Financial Services granted Ripple a license for the virtual currency “XRP”, allowing businesses, banks, and regular consumers to send and receive money internationally.

As it is considered the basic settlement infrastructure for interbank transactions by linking the various interoperation ledgers and payment networks of banks and nonfinancial services companies by integrating the Ripple platform into their own systems, and thus allowing its customers to use its settlement services such as direct and immediate transfer of money between two parties; any form of money, including Dollars, Euros, Yen, and Gold, can be exchanged; similar to Bitcoin, Ripple also uses its own type of virtual money to enable lowcost money transfers between financial institutions, as one of XRP's unique uses is as a bridge currency, which may be required if two currencies cannot be exchanged directly at a given moment[1].

5. Blockchain structure

Blockchain is a secure storage technology. It is composed of a chain of chronological blocks, each holding a list of valid transactions in the network used for transfer, collaboration and coordination. Generally, blockchain is structure using terms like ledger, transactions, chain, blocks [11].

5.1. The Chain of Blocks

The term “chain” or “blockchain” in the context of blockchain refers to a cryptographic chain of blocks, with each block containing a unique hash as well as the hash of the previous block. This chain forms a secure sequence, or blockchain. Each block contains a hash (a cryptographic signature) generated from the data it contains. The next block in the chain includes this hash in its own header, thereby making its own hash dependent on the previous block. This ensures that any modification to a block’s data will alter its hash, thereby breaking the chain and invalidating all subsequent blocks. Consequently, altering the data in a block requires recalculating the hashes of all subsequent blocks, which is computationally impossible, thereby ensuring the immutability and security [12].



Figure1. 7: Sequential Block Linking Mechanism[13].

5.2. Block

A block is the fundamental unit of the blockchain that stores a batch of transactions. Each block contains two components, block header and palyoad [14].

5.3. Block Header

Contains metadata about the block,[11]the most important elements:

hash:The hashes are a fundamental brick of blockchain technology. There are used to prove the integrity of the blocks and create a link between consecutive blocks that built the chain. Hence, it is possible to exemplify the use of the hashes as the digital fingerprint of each ledger page (the block) and the glue that assembly each ledger page (the chain).

MerkleTrees: Merkle trees are a type of binary tree composed of a set of nodes with the leaf nodes representing data blocks. Non-leaf nodes are representing the cryptographic hash of their offspring. This structure allows for efficient and secure verification of large sets of data. The use cases are efficient data verification and forming the block structure in some Blockchain architectures [4].

Nonce: A 4byte field that usually begins at zero and increases with each hash calculation.

Timestamp: Produces timestamp with the current time as seconds for every block[3].

5.4. Block Body

La partie centrale du bloc contient une liste de transactions. Chaque transaction correspond à un transfert de valeur ou à l'exécution d'un contrat intelligent, et comprend des informations telles que l'expéditeur, le destinataire, le montant et les signatures numériques permettant de vérifier son authenticité.

5.5. Consensus algorithm

When a block is to be added to the blockchain, it must be validated by all nodes in the network. Consensus algorithms are a type of protocol that allows nodes participating in the blockchain network to agree on the order of transactions and filter out invalid ones. Multiple transactions compete with one another to be published and receive the transaction reward. Consensus algorithms were created to solve this decision-making

problem. It is also difficult for miners to reach a consensus because, in the absence of a central authority, the blockchain network is distributed[3].

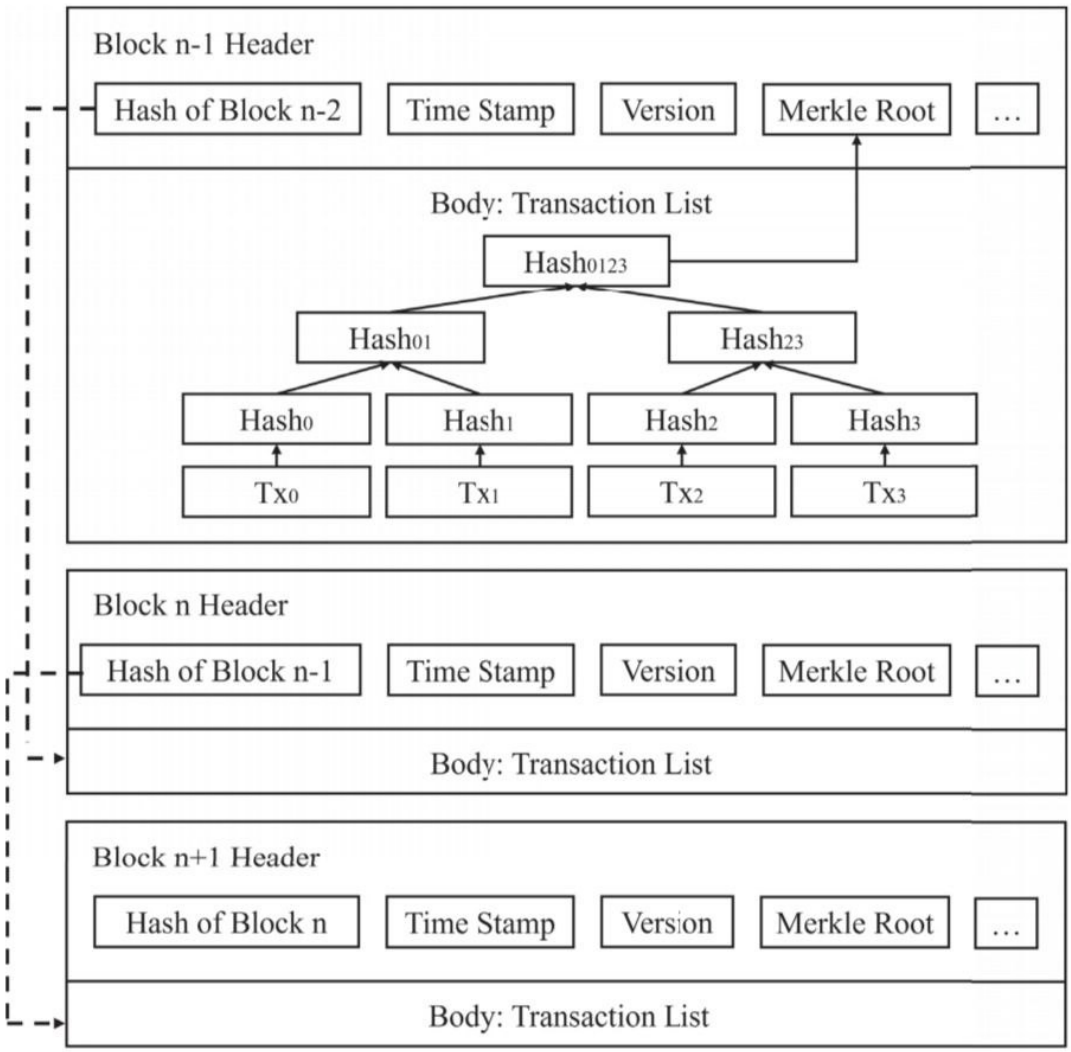


Figure1. 8: Block Structure[14]

6. Advantages of Blockchain

The following points highlight the main advantages offered by blockchain:

- Faster Transactions: Blockchain significantly reduces transaction processing time from several days to just a few minutes or seconds [6].

- Traceability: Every transaction is permanently recorded, making it easy to verify and track the complete transaction history [15].
- User Autonomy: Users have full control over their data and transactions without relying on centralized authorities.
- Reliability and Durability: Because blockchain is decentralized, it continues to operate even if some nodes fail, ensuring high availability.
- Cost Efficiency: Blockchain reduces transaction costs by eliminating intermediaries, lowering fees, and enabling faster, more efficient cross-border payments [1].

7. Blockchain Challenges

Emerging technologies such as blockchain face several challenges that are slowing their adoption. Sharma identified five major challenges to blockchain adoption in 2020: scalability, energy consumption, latency, interoperability, as well as a lack of skills and standardization[16].

- **Scalability :**According to the Canadian Institute of Chartered Accountants, scalability refers to the ability of a product or system to maintain its functional properties despite changes in size or certain parameters. One of the main limitations of current public blockchains is their lack of scalability[17]. This concept was initially described by Vitalik Buterin, the co-founder of Ethereum, who stated that tradeoffs are inevitable among three key properties of blockchain: decentralization, scalability, and security. Decentralization is the core and essence of blockchain. Security is an essential property, while scalability is the main challenge. In other words, the scalability trilemma indicates that trade-offs are almost inevitable among these characteristics of the blockchain. These properties are negatively correlated. Thus, by focusing too much on two of them, the third will be negatively affected[6].

- **Energy consumption:** Maintaining a secure system entails significant costs, particularly in terms of energy consumption. It is estimated that the current Bitcoin ecosystem consumes as much electricity as some small towns. In the context of Internet-scale networks, which encompass multiple types of connections and devices, such as mobile phones, energy consumption becomes a critical issue. This requires intelligent management of data and computations, based on each device's processing power and battery life.
- **Standardization:** There is currently no standardized framework for assessing the compatibility of blockchain technology with existing laws and regulations, nor for integrating it into existing systems. This lack of standardization and clarity hinders the widespread adoption of this technology[11].
- **Latency and finality:** First, grouping transactions into blocks introduces significant latency in information exchange, which limits potential applications to use cases requiring real-time transfers. In a blockchain context, we define latency as the time between when a user initiates a transaction and when other users consider that transaction to be part of the blockchain. This latency depends on the block frequency of the blockchain in question, but also on the number of confirmations required to ensure network security. The average block creation times for Bitcoin and Ethereum are approximately 10 minutes and 14 seconds, respectively, resulting in very high latencies. We therefore define the security of a blockchain as its ability to ensure the immutability of transactions made on the network.
- **Number of Transactions and Costs:** Another limitation is the raw number of transactions the network can process in a given time period, which corresponds to the number of transactions that can be included in a block multiplied by the block generation interval. Thus, if too many users attempt to conduct transactions simultaneouslyrelative to the network's capacity, transaction fees increase,making certain potential applications requiringa large number of exchanges impossible. For example, the Bitcoin blockchain has atheoretical limit of 7 transactions per second for the entire network [17].

8. How Blockchain Works?

Following picture depicts how Blockchain works

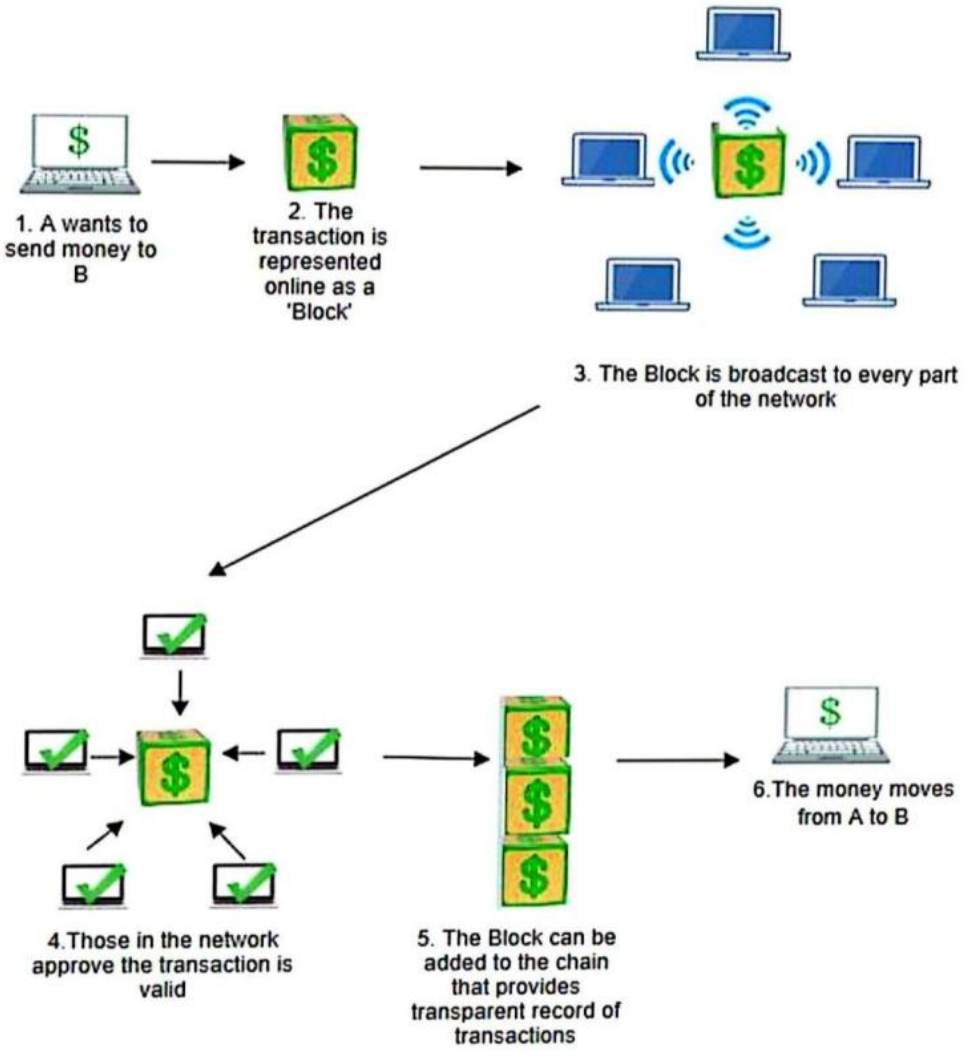


Figure1. 9:Blockchain worksStructure[18].

9. Blockchain Security

Blockchain security refers to the use of cryptographic techniques, consensus mechanisms, and decentralized architectures to ensure the integrity, availability, and confidentiality of data stored on a blockchain network. It involves protecting the system against unauthorized access, tampering, and cyberattacks, while maintaining

trust among distributed participants without the need for a central authority. Blockchain systems rely on secure hashing algorithms, digital signatures, and network protocols to prevent fraud and ensure that once data is recorded, it cannot be altered without detection[19].

10. Security aspects in Blockchain networks

Security in Blockchain networks is provided by means of advanced cryptographic techniques,various methods for determining consensus and, above all, the core aspect of this technology, i.e. immutability of data. Joshi et al. in defined several important security principles that should be followed when building and supporting a blockchain network:

- **defence in penetration:** the use of multiple layers of security is more effective than the application of a single layer.
- **minimum privilege:** access to data should be limited to the lowest possible level,
- **manage vulnerabilities:** security vulnerabilities ought to be constantly checked and corrected.
- **manage risks:** the risk should be regularly assessed and managed at an environmental level.
- **manage patches:** faulty system components should be corrected, tested and developed as patches for the successive versions of the application.

Blockchain systems are based on many techniques that ensure an adequate level of security. In practice, consensus models are responsible for providing the appropriate level of network safety. For example in PoW to make the network fake, the node would have to possess at least 51% of the computing resources of the entire network to be able to falsify the information contained in the blocks, which is hardly possible in practice [20].

11. PRIVACY AND SECURITY TECHNIQUES USED IN BLOCKCHAIN

In this section, we provide a detailed discussion on a selection of techniques that can be leveraged to enhance the security and privacy of existing and future blockchain systems.

11.1. Anonymous Signatures

Digital signature technology was developed by several variants. Some signature schemes themselves have the ability of providing anonymity for the signer. We call this kind of signature schemes anonymous signature. Among the anonymous signature schemes, group signature and ring signature were proposed earlier and are the two most important and typical anonymous signature schemes.

11.1.1. Group Signature

Group signature is a cryptography scheme proposed initially in 1991. Given a group, any of its members can sign a message for the entire group anonymously by using her personal secret key, and any member with the group's public key can check and validate the generated signature and confirm that the signature of some group member is used to sign the message. The process of signature verification reveals nothing about true identity of the signer except the membership of the group.

Group signature has a group manager who manages adding group members, handling the event of disputes, including revealing the original signer. In blockchain system, we also need an authority entity to create and revoke the group and dynamically add new members to the group and delete/revoke membership of some participants from the group.

Since the group signature requires a group manager to setup the group, the group signature is suitable for consortium blockchain. Recently, PlatON added group signature in its platform for providing users with anonymity support.

11.1.2. Ring Signature

Ring signature also can achieve anonymity through signing by any member of a group of users. The term of “ring signature” originates from the signature algorithm that uses the ring-like structure. The ring signature is anonymous if it is difficult to determine which member of the group uses his/her key to sign the message. Ring signatures differ from group signatures in two principal ways: First, in a ring signature scheme, the real identity of the signer cannot be revealed in the event of dispute, since there is no group manager in ring signature. Second, any users can group a “ring” by themselves without additional setup. Thus, ring signature is applicable to public blockchain. One of the typical applications of ring signature is CryptoNote. It adopts ring signature to hide the connection between the sender’s addresses of transactions. More precisely, CryptoNote constructs the sender’s public key with several other keys so that it is impossible to identify who actually sent (signed) the transaction. Due to the use of ring signature, if the number of ring members is n, then the probability that an adversary may successfully guess a real sender of a transaction is 1/n. Later, Ethereum added ring signature in 2015, which gives the users anonymity like CryptoNote currencies such as Monero.

11.2. The Trusted Execution Environment (TEE) Based Smart Contracts

An execution environment is called TEE if it provides a completely isolated environment for application execution, which effectively prevents other software applications and operating system(s) from tampering with and learning the state of the application running in it. The Intel Software Guard eXtensions (SGX) is a representative technology to implement a TEE. For example, Ekiden is a SGX-based solution for confidentiality-preserving smart contracts. Ekiden separates computation from consensus. It performs smart contract computation in TEEs on compute nodes off chain, then uses a remote attestation protocol to validate the execution correctness of compute nodes on chain. The consensus nodes are used for maintaining the

blockchain and do not require to use a trusted hardware. Enigmautilizes TEE in its current version to allow users to create privacy-preserving smart contracts using a decentralized credit scoring algorithm. Multiple factors are weighted for credit scoring, such as the number and types of accounts, payment history, and credit utilization [21].

11.3. Cryptography

Cryptography or the art of hiding messages has become nowadays a science onits own; it combines multiple disciplines such as mathematics, computer sciencesand even physics to ensure the protection of the data by providing several securityservices such as authenticity, integrity, condentiality and non-repudiation. Thereare two main cryptographic schemes [22]**:**

11.3.1. Symmetric Cryptography

Symmetric cryptography, is a setting where two parties share some secret information called a key, and use this key when they wish to communicate secretly with each other. An implicit assumption in any system usingprivate-key encryption is that the communicating parties have some way of initially sharing a key in a secret manner. Two users share a secret key K to communicate confidentially. Each user encrypts the message he wants to send to the other user, using the key K, and the other user decrypts the received encrypted message to get the original message back, using the same key K. Anybody who intercepts the ciphertext should not be able to learn anything about the original message, without knowledge of the secret key K[23].

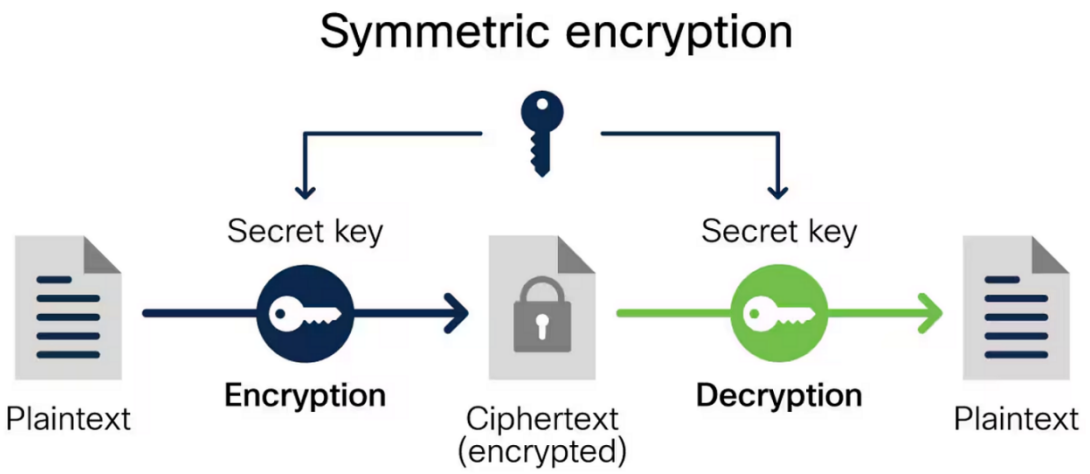


Figure1. 10: Symmetric Cryptography[24].

11.3.2. Asymmetric Cryptography

In contrast to symmetric cryptography, in asymmetric cryptography (also called public-key cryptography) the two parties do not share the same secret key K but they use two different keys, a public key (PK) and a secret key (SK). An important property is that given the public key it is infeasible to forge the secret key. In this scheme, the receiver sends his public key to the sender over an insecure channel, who then encrypts the plaintext using that key and transmits the resulting ciphertext. Finally, the receiver can decrypt the ciphertext using his secret key. In Public Key Cryptosystems both users exchange their public keys publicly to encrypt a plaintext that can be decrypted only with the secret key associated with each public key. By publicly revealing PK one does not reveal an easy way to compute SK[23].

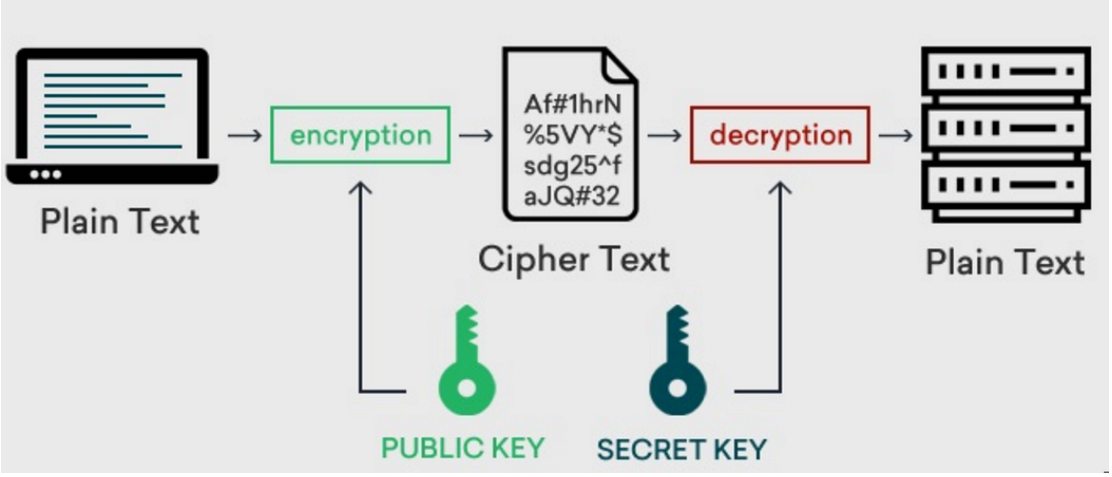


Figure1. 11: Asymmetric Cryptography[24].

12. Blockchain as a security solution

The main reason why blockchain can be used in network security technology is that it has high security. It is embodied in two aspects:

12.1. Data usage security

The decentralized data management system enables the blockchain system to effectively connect with external information data. In the case of separation of permissions, it is more convenient for us to extract relevant information, which greatly reduces the probability of security risks in central institutions. Also according to (Liu and Xu, 2018), the blockchain system has very good data protection performance, so that the information in the blockchain cannot be tampered with, which further improves the security performance of network data information. When using a program to access related data, you must first obtain the user's access rights. Before storing data information, the blockchain will first encrypt the data information in a relatively safer way, and then enter the system for storage. After completing the security settings and storage, if the user wants to change some data authorization information again, thepermission settings need to be updated to determine the original information technology permission settings stored in the blockchain.

Nodes' access to the blockchain system needs to issue an access request in advance, and make accessible records for comparison between the system and the signature before access, so as to open the authorized node information for users to enable them to smoothly access information data. Using this technology, the node's access to the data information in the system will be preserved throughout the process, and the node's utility is to change the corresponding data access information authority. Users access the required information data with the help of the blockchain system. They can be fully displayedin the network system so that other users can see that through these records. All systems users with corresponding permissions can audit access records.

12.2. System Security

When the blockchain system asset is designed, it needs to be encoded. In the process, some security vulnerabilities will inevitably appear, and some 22 network viruses will attack these security vulnerabilities, thus forming a blocking effect on the effective application of the blockchain. Based on the characteristics of the blockchain system itself, viruses can only attack a single node, and it is difficult to cause a large-scale impact on the entire blockchain system. Therefore, under the function of blockchain, the blocking effect can effectively improve the security and defence performance of the network and provide userswith a relatively safe network environment [25].

CONCLUSION

Blockchain is a revolutionary concept as it has been successfully able to bring the transparency among the users and has become a game changer for many industries. Blockchain encourages entrepreneurship by destroying Therefore, in the next chapter, we will explore the concept of Cloud Computing to understand its principles and analyze how its integration with Blockchain can offer even more efficient and secure solutio

Cloud computing

Chapitre 02

Chapitre02: Cloud computing

Introduction

Cloud Computing has emerged as one of the most significant technological innovations, enabling organizations and individuals to access computing resources, storage, and services over the Internet in a flexible and cost-effective manner. Its widespread adoption has transformed the way data is stored, processed, and managed. However, despite its numerous advantages, cloud computing faces several security and privacy challenges that may affect users' trust and data protection.

This chapter presents the fundamental concepts of Cloud Computing, including its architecture, models, characteristics, and major service providers. It also discusses the main security concerns and protection mechanisms used in cloud environments. Finally, a comparison between Cloud Computing and Blockchain technology is introduced, along with an overview of how blockchain can be integrated with cloud systems to enhance security, transparency, and reliability.

1. Cloud Computing

1.1. History

There is no specific date that can be pinpointed as marking the birth of the Cloud. The concept of the Cloud dates back to the 1950s, when large-scale mainframe computers first appeared in universities and businesses and became accessible via client terminals and computers. As early as the 1960s, John McCarthy believed that

"computing might one day be organized as a public utility". Furthermore, the usage characteristics of the public and private sectors such as the electricity sector, government agencies, and community organizations explored in Douglas Parkhill's 1966 book, bear many similarities to the modern characteristics of the Cloud.

Since the 1970s, the term "business service provider" has been used to describe a company that rents out telephone lines, answering machines, IT services, and so on. Generally, business service providers' clients lack both the capacity and the expertise to integrate these services in-house, which is why they turn to a provider. "Application Service Providers (ASP)" also play a role in the history of the Cloud. An ASP type entity offers an application provided as a service this is what we now refer to as SaaS, or "Software as a Service," in current Cloud terminology. Furthermore, the term "Cloud" was already in use as early as 1990 to refer to large ATM (Asynchronous Transfer Mode) networks, which rely on the concept of virtual circuits and encompass software, hardware, and connection media, all while supporting multiple levels of service quality.

In the 21st century, the term "cloud computing" began to emerge. In 2002, Amazon, the leader in e-commerce, pioneered the concept of the cloud. In fact, they invested in a cluster of massive servers, designed to handle the heavy load of orders placed on their site during the Christmas season, but largely unused the rest of the year. Their idea was therefore to make all these unused resources available to businesses, so they could rent them on demand. Subsequently, many companies entered the cloud market, such as IBM, Microsoft, and Google. The year 2007 saw an increase in activity among these companies (Google, IBM, etc.). Subsequently, a number of universities became heavily involved in large-scale research projects focused on the cloud, and over time, the term began to gain popularity in the press [26].

1.2. Définition du Cloud Computing

Cloud computing is a modern information technology model that enables on-demand access to computing resources such as servers, storage, databases, and software over the internet. These services are delivered through large data centers and can be rapidly scaled and accessed anytime without requiring local infrastructure. It provides organizations with cost efficiency, flexibility, improved collaboration, and high availability of services, while also introducing challenges related to security and data privacy [27].

1.3. Characteristics of cloud Computing

Generally, a service, solution, or execution environment must meet a set of characteristics to be considered cloud computing. Among these characteristics, some are recognized as fundamental. For example, NIST defines five essential characteristics, which are :

- **Ondemand resources:** A user can unilaterally allocate computing resources (servers, network, storage, runtime environment, application) as needed, automatically and without the need for human interaction with each service provider.
- **Broad network access**: Cloud computing resources are available over the network and accessible via standard mechanisms that facilitate their use from heterogeneous client devices, including lightweight ones.
- **Resource sharing:** The cloud computing provider's computing resources are shared to serve multiple clients using a multi-tenant model. These resources, whether physical or virtual, are dynamically allocated and released based on consumer demand. Generally, the user has neither control over nor knowledge of the exact location of the allocated resources. In some cases, the user may choose the geographic location at a high level.
- **Rapid elasticity:** Resources are allocated and released elastically ideally automatically to adapt quickly to demand, whether it is increasing or decreasing. To

the consumer, the resources available for allocation appear to be unlimited and can be allocated at any time.

- **Metered services:** All allocated resources can be monitored and controlled to measure their consumption at an appropriate level of abstraction depending on the type of service (e.g., storage, compute time, bandwidth)[28].

1.4. Cloud Computing Architecture

Looking at the literature, there is no standardized or universal cloud architecture reference model available. The cloud computing industry and academia are working in parallel to define and standardize reference models, frameworks and architectures for cloud computing. Therefore, cloud reference models are in the development stage, and are expected to stay that way as long as the cloud computing models are evolving. According to, Forrester Research was able to count more than 75 different parties working on cloud standards in 2010. Today, the number of industry and academia involved has clearly increased significantly. From the literature, however, cloud computing architectures are better classified by service and deployment models. In the next sections most common cloud computing service and deployment models are discussed[29].

1.5. Cloud Models

1.5.1. Deployment models

there are four primary ways in which cloud services are deployed. Cloud integrators can play a vital role in determining the right cloud path for a specific organization [30].

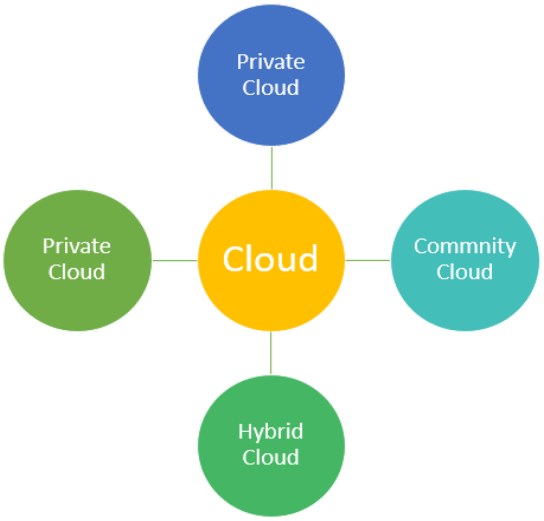


Figure2. 1:Cloud Cloud Models.

- **Private cloud:** the cloud infrastructure is dedicated to only a single organization. Resources are maintained inhouse by an organization, or are allocatedexclusively for an organization by a cloud provider on its premises.
- **Community cloud:** the cloud infrastructure is dedicated for exclusive use by aspecic community of users from two or more organizations that have sharedinterests. Resources are maintained in-house, or are allocated exclusively forthe community by a cloud provider on its premises.
- **Public cloud:** the cloud infrastructure is provisioned for open use by the gene-ral public. It may be owned, managed, and operated by a business, academic,or government organization, or some combination of them, on the premises ofthe cloud provider.
- **Hybrid cloud:** is a cloud infrastructure which is composed of two or moredistinct private, community or public cloud infrastructures that remain in-dependent entities, but are bound together by standardized or proprietarytechnology that enables data and application portability (e.g., cloud burstingfor load balancing between clouds) [31].

1.5.2. service models

1.5.2.1. Classic Cloud service models

- **Infrastructure as a service:** IaaS is the infrastructure service model that includes storage and virtual machines. Load balancing in cloud computing is performed

using IaaS. Users can install access to required software through virtual machines. These virtual devices provide on-demand facility to the customers. The IaaS service offers hardware platform to the users on-demand basis. Therefore, users can access the online hardware platform as on-demand basis to fulfill their requirements. Additionally, the IaaS service also supports virtualization of resources on which a guest user can run his/her own operating system.



Figure2. 2: LaaS Infrastructure Model[32].

- **Platform as a service:** PaaS is responsible for the development and delivery of programming models to IaaS. Users can access such programming models through cloudand execute their programs. PaaS is responsible for the runtime execution of users' given task. Therefore, the PaaS service completes the requirements of building and delivering of Web-applications without downloading and installing required software as well.

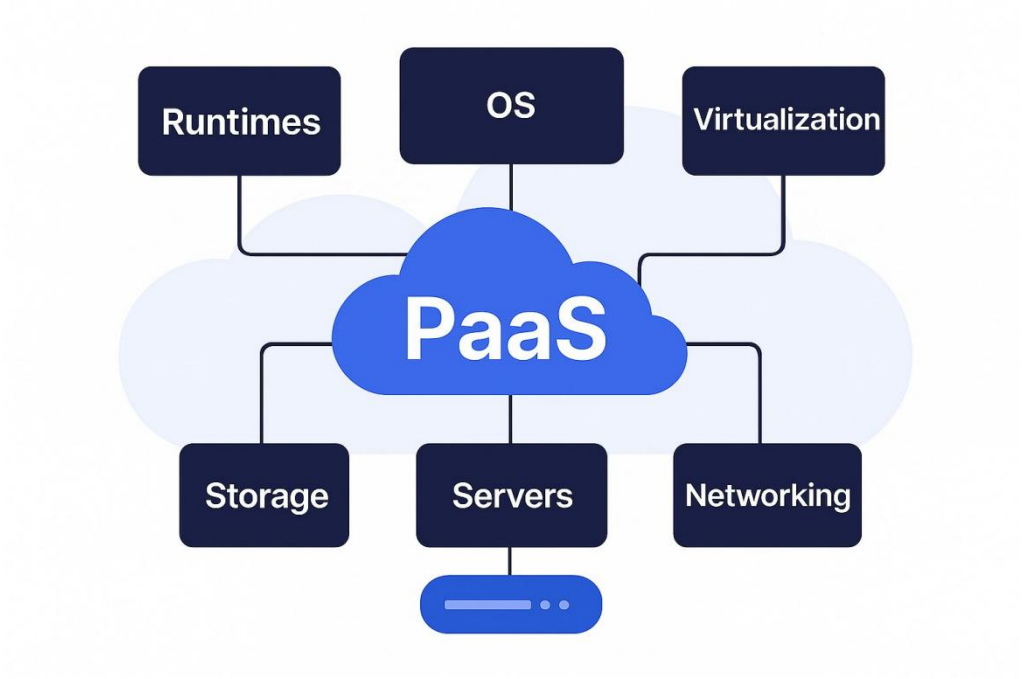


Figure2. 3: *PaaS Infrastructure Model[32]* .

- **Software as a service:** SaaS supports all the applications in the cloud environment. This feature of cloud computing is accessible through Web-browsers. The SaaS service providesthe modeling of software deployment where users can run their applications without installing it on his/her own computer. However, this service is limited to the users, i.e., only existing set of services is available to the customers[33].

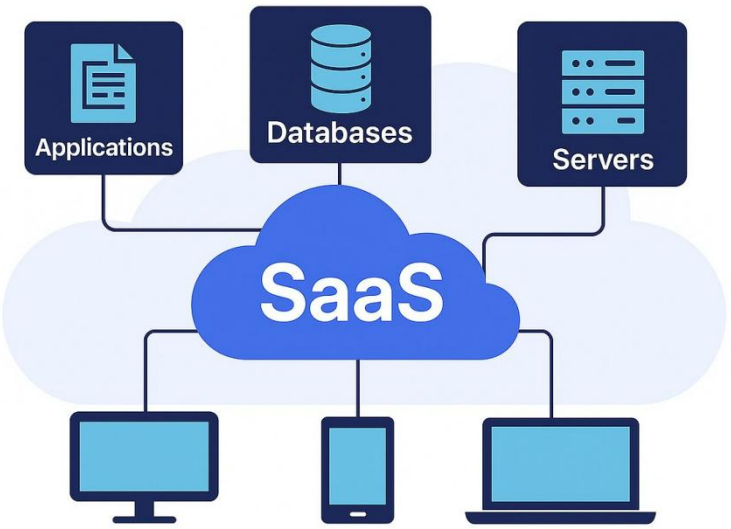


Figure2. 4: SaaS Infrastructure Model [32].

1.5.2.2. New hybrid service models

As Cloud is maturing and as users are requesting more flexibility and more controlsince they need to deploy their applications with the topology they choose and withhaving the control on both infrastructure and programs, the classic three layerconcept has often been a subject of speculation and discussion. Lines between Cloudservices are blurring which results in the combination of IaaS and PaaS and tothe appearance of new hybrid Cloud providers that enable users to create a singlecontinuum of services.

Leading companies like Amazon, Microsoft and Google are concurrentlygoing to blend IaaS and PaaS and don't want users to think strictly about IaaS orPaaS when they require Cloud services. An example of this trend of new combinedCloud services is Kubernetes. It is a newly released Google solution that solvesboth IaaS and PaaS. Openstack IaaS provider is also gaining PaaS featuresand providing combined IaaS-PaaS services by orchestrating Docker containers via Openstack Heat. Details about this mechanism are given in Appendix B.451 Research, a leading global analyst and data company focused on the busi-ness of enterprise IT innovation, states that: " Although it is maturing in tech-nology and market, PaaS is getting squeezed between

consolidation with IaaS andheavy use of SaaS. PaaS will most likelysurvive as a category, but not necessarilyas we know it today"[34].

1.6. Cloud Platform

1.6.1. Amazon Web Services (AWS)

Since its launch in 2006, Amazon Web Services (AWS) has been a pioneer in the cloud computing industry. Offering a range of services to meet various computing demands such as; storage, database management, analytics, network, Internet of Things (IoT), mobile computing, and enterprise solutions.

These services help organizations to escalate their growth, reduce costs, and scale their operations. Using its expertise and resilient infrastructure, AWS has secured its position as a dominant cloud platform worldwide. The global presence of AWS operates in 105 availability zones in various regions. This vast global reach guarantees reliability and redundancy, allowing businesses to deploy their applications and services securely. Based on its presence and relentless innovation, AWS became the go-to platform for organizations looking for cloud solutions to support their digital transformation efforts.

1.6.2. Google Cloud Platform (GCP)

Google Cloud Platform (GCP) emerged in 2011 as Google's venture into the realm of cloud computing, offering an array of services to meet the diverse needs of its customers. With a robust suite of offerings encompassing storage, big data solutions, databases, analytics, cloud AI,International Journal ofComputing and Digital Systems networking, mobile computing, development tools, management tools, Internet of Things (IoT), cloud security, and data transfer, GCP provides a comprehensive ecosystem for organizations seeking scalable and innovative cloud solutions .

GCP's inception marked a significant milestone in Google's commitment to delivering cutting-edge technology services to its user base. Google Cloud coversavailability zones strategically distributed across the globe. This large network

zonespresence ensures high availability and reliability, allowing businesses to deploy their applications easily while leveraging Google's robust infrastructure. GCP's dedication to providing a secure, scalable, and agile cloud platform makes it competitive player in cloud computing landscape, catering to the evolving needs of businesses across industries.

1.6.3. Microsoft Azure (Azure)

launched in 2010, represents Microsoft's comprehensive cloud computing platform designed to provide a wide array of services to businesses and developers. Azure offers a diverse range of cloud services, including computing, storage, databases, analytics, networking, machine learning, AI, IoT, security, and developer tools. As one of the leading cloud platforms, Azure enables organizations to build, deploy, and manage applications and services with flexibility, scalability, and reliability.

With a global presence spanning over 60 regions and 140 countries, Microsoft Azure provides extensive coverage and high availability to meet the diverse needs of its users worldwide. The platform's integration &with Microsoft's extensive suite of enterprise products and services further enhances its appeal, allowing seamless interoperability and enabling organizations to leverage their existing investments in Microsoft technologies[35].

1.7. Cloud Providers

According to NIST, there are five key players at the heart of the cloud computing paradigm: the provider, the user, the auditor, the broker, and the cloud carrier. These players represent a physical or legal entity (an individual or an organization) that participates in cloud-based processes or transactions, as follows:

- **cloud provider:** Represents a person or organization that makes a set of cloud services available to potential users (or consumers).
- **cloud customer:** Represents a person, organization, or entity that uses the various services made available by the provider, generally through a customer contract.

- **cloud auditor:** Represents an entity whose task is to measure and evaluate cloud services in terms of performance, security, costs, etc., while remaining independent of the provider and cloud users.
- **cloud broker:** Represents a party responsible for managing and providing cloud services by negotiating relationships between users and the cloud provider.
- **The cloud carrier :** Represents an organization or intermediary entity responsible for providing connectivity and ensuring the availability of cloud services by transporting them from the provider to the consumer[36].

1.8. Advantages of Cloud Comiting

Cloud computing has been around for about two decades. In addition to data demonstrating its effectiveness and the cost and competitive advantages it offers over traditional operating models, cloud computing has other strengths, including:

- **Cost-effectiveness:** Cloud computing is likely the most cost-effective method to use, maintain, and upgrade. Traditional desktop software is expensive for businesses to finance. Additionally, the extra costs of licensing applications for multiple users can become very costly for any organization.
- **Unlimited storage:** Storing information in the cloud provides virtually unlimited storage capacity. Consequently, there is no longer a need to worry about running out of storage space or expanding current storage capacity. There are hundreds of different storage systems available. Some serve very specific purposes, such as storing emails or digital images. Others are available to store all forms of digital data.
- **Automatic software integration:** In cloud computing, software integration is typically automatic. This means there is no need to go to extra lengths to customize and integrate applications according to your preferences. Furthermore, cloud computing greatly simplifies the customization of options. You can also select only the services and software applications that best suit your organization.

- **Easy access to information:** Once you've signed up or registered for cloud computing, you can access information from anywhere, provided you have an internet connection. This convenient feature allows you to overcome geographical constraints[37].

- **Savings:** It is the most frequently mentioned benefit. Cloud Computing lowers the cost for using IT services and the cost of everything that has to do with IT maintenance. It reduces the cost of acquiring and maintaining computing power and allows the organizations to buy only the computing services needed. It also reduces the purchase and updating cost of the software that an organization or an end-user needs. The Cloud services may be acquired for a specific period of time. This enables agencies to not invest in complex and expensive IT infrastructures which may be used only temporarily.
- **Flexibility:** The Cloud enables users to adapt the services they use according to their needs. Companies may start with a small amount of resources and grow to a large amount of resources rapidly, and then scale back if necessary. Furthermore, the companies can adapt to the users demands over a period of time; if the users demands grow, companies may extend their resources and then release them after the demands fall.
- **Maintenance:** The maintenance is made by cloud providers. A customer company/user(does not need to worry about the hardware, the) operating systemor the software maintenance. The maintenance cost is included into the cost of the service provided. This gives the customer the impression of a free maintenance.
- **Availability:** From a customer point of view, cloud services are always available. she/he does not need to worry about the management of failures that may occur while a certain service is used. The Cloud providers are required to manage any type of problem that may arise. Furthermore, Cloud Computing can be utilized as a viable disaster recovery option-especially for storage.
- **Efficiency:** The maintenance made by Cloud providers may increase the overall

efficiency of the system.

- **Mobile Access:** Mobile users can benefit from the services provided by Cloud Computing. The reduced storage and processing capacities that characterized till now the mobile device were an impediment into the use of various tasks and applications. Now, through Cloud services, mobile users can take advantages of external IT services and use various applications [38].

2. Cloud Security

2.1. Need for Data Security in Clouds

The basic requirement of data protection is security and it must be implemented by cloud computing system specifically or generally by the Information Technology System. The cloud service providers perform security mechanisms for user needs. There are three main requirements for security which are confidentiality, integrity, and accuracy of data.

Cloud computing is the most oriented field of information technology. Independent user's social media and its related structures shifting towards the outstanding filed of cloud computing. Therefore, cloud goes through know of any issues. In the development of data these issues cause problems. In cloud computing data is stored in remotes which are note under the users control. So, the data on cloud computing is approach from everywhere and it is easily susceptible. Still the remotes server is not trustful and easily cause misuse and delete of data. Even through cloud architectures has its own security, so it is essential.

Typical cloud computing has particular security architecture along many customers with multiple demands. In this situation, adjustments on services of cloud computing is not possible. It makes sure data should not be manipulating by cloud server. In cloud computing storage of data and its security is ensured by the core security. So, it is essential to compute security proceeded at the end of users. Most people believe that data is safe in local storage, but some others believe that external data is more

safe because external companies are more responsible for data and they do not have any concern with your data[39].

2.2. Security issues in the cloud

According to the National Institute of Standards and Technology, cloud computing is amodel for enabling ubiquitous, elastic, on-demand network access to a shared pool of config-urable computing resources (e.g., networks, servers, storage, applications, and services). Itsservice models, known as Software as a Service (SaaS), Platform as a Service (PaaS) andIaaShave specific and shared security challenges. The first provides a CSC the capability touseapplications running on a cloud infrastructure by a CSP. PaaS provides the CSC with toolsto deploy their applications on top of the cloud infrastructure. The latter gives provision tothe CSC in processing, storage, networks, and other fundamental computing resources wherethe consumer can deploy and run arbitrary software, which can include operating systems andapplications. Security challenges reside in the coexistence since PaaS, as well as SaaS services, are hosted on top of IaaS. Therefore, any threat in IaaS can impact the security of both, PaaSand SaaS, and vice versa.We further enumerate critical cloud aspects trying to provide the explanations how thes eelements can influence the design of a cloud-based threat detection system. In particular, we consider:

- **Virtualized environment** Brings elasticity by allowing multiple Virtual Machine (VM) management and pooling in the same physical resources.
- **Multi-tenancy** Enables the use of a single resource by multiple customers that may or may not belong to the same organization.
- **Data life cycle** Defines no fixed infrastructure and security boundaries on applications and data on the cloud.
- **Network dynamics** Concerns non-linear, non-stationary and complex dynamicalcharacteristics of the network flows.
- **Access** Takes into account the fact that data are transmitted using the Internet and may require credentials, authentication, identity management and

anonymization[40].

2.3. CLOUD COMPUTING SECURITY CONCERNS

Regulatory Compliance: Even if a cloud claims to be secure enough from attacks, Clients are using cloud services eventually at their risks only. Clients need to make sure that the cloud provider they are dealing with is accredited or certified with a good security program. For example, in healthcare industry, it is required for the cloud provider and the development and support team employees, who work on the data to take a compulsory HIPPA (Health Insurance Portability and Accountability Act) test. This type of certification helps the clients to be assured that their information is safe and not lost. Few other examples of standard accreditation and certification are ISO 27002, Safe Harbor, ITIL (Information Technology Infrastructure Library), and COBIT (Control Objectives for Information and Related Technology).

Audit permission: Sometimes the clients might want to have an external party or cloud for auditing to make sure their data is protected. This requires that the cloud providers agree to the term of audit.

Employees: The people working in a cloud provider company are the most crucial point of security concerns. Most of the time, attackers are insiders who know everything about the system. The clients should have right to know how the provider companies are taking care of this issue[41].

Account or Service Traffic Hijacking: This includes attacks like phishing, fraud, and exploitation of software. This security threat arises due to loss of credentials. If the attackers gain access to your credentials and passwords, they can easily get access to your confidential data and accounting information. Also they can manipulate or delete your data, return falsified information, and redirect your client to some illegitimate site. To avoid this, the organizations or individuals should protect their passwords and credentials from being shared between users and services. Also proper authentication technique should be used.

Insecure Interfaces and APIs: Sometimes customers have to interact with the

service providers. For this the providers expose a set of software interfaces or APIs. Management, controlling are done using these interfaces. The security of these APIs is very important. Authentication and encryption is used to make them secure. Also they must be designed to protect against both accidental and malicious attackers.

Denial of Service(DoS) : attack won't let the users of cloud services to access their data or applications. The attacker (or attackers as is the case in Distributed Denial of Service (DDoS) attack) creates a system slowdown by consuming system resources such processor power, memory, disk space, bandwidth. This creates confusionamong the users and they get annoyed over the late response for their requests. DoS attack leaves the users into an unending loop in which they are not able to do anything except to sit and [42].

2.4. Fundamentals of security

Information systems security is commonly defined as the protection of information and information systems against unauthorized access, use, disclosure, disruption, modification, or destruction, ensuring the preservation of confidentiality, integrity, and availability (CIA). Based on this definition, three main security services can be identified:

- **Condentiality:** which means protecting information from disclosure tounauthorized parties.
- **Integrity:** which refers to protecting information from being modified byunauthorized parties.
- **Availability**: which means that the data should be available all the time andauthorized parties should be able to access the information when needed.In addition, we find other security services that might be important in some scenariosand applications, such as:
- **Authentication:** which refers to the verication of users' identities.
- **Access control:** which refers to providing protection against unauthorizeduse of resources.

- **Non-repudiation:** which refers to the non-denial of any action performed byany user in the system.
- **Privacy:** which refers to the protection of personal information [22].

2.5. Security mechanisms

Security mechanisms are the technical tools and techniques that are used toimplement security services. A mechanism might operate by itself, or with others,to provide a particular service. In this section we focus on two security mechanismsthat we present in what follows [22]:

2.5.1. Access Controls and Authentication

It is crucial for technology firms to always focus on access controls of private databases. In this regard, an authentication procedure is utilized, which helps in protecting personal data. Strong passwords and other security protocols that cannot be easily breached are used in such cases. The use of access controls assists in limiting the number of people using a specific dataset. The fewer people use the data, the fewer the chances of breaches. This concept is very important to understand in the area of cloud warehousing. Only a single breach can put the firm's reputation at stake. Companies should also use the authentication technique to secure their reputation in the industry. In this case, verification is always required when someone accesses the private data. With verification, access is granted to everyone. The verification can be done through different methods like strong passwords, biometrics, etc. All such measures add a level of security to the cloud data.

2.5.2. Auditing and Monitoring

Companies involved in cloud data warehousing should also be aware of auditing and monitoring procedures. These methods help implement transparency in security measures and detect areas of weakness. Regular auditing and monitoring procedures are crucial in identifying areas with weak security. In this case, strong security measures are immediately implemented in such areas to avoid any future breaches.

- **Comprehensive Audit Trails:** As discussed earlier, detailed audit trails help

companies track every single action taken related to the use of cloud data. Even a slight modification in sensitive data is detected immediately, and audit trails are conducted to identify the problematic areas. In this way, security violations are immediately detected, and strong security measures are put in place on time before any huge breaches.

- **Real-Time Monitoring:** Companies should also implement real-time monitoring of cloud data warehousing. If any unauthorized access occurs to sensitive data, it should be detected and monitored in real-time. The patterns of data access should be regularly monitored to identify the presence of hackers anywhere in the cloud.
- I**ncident Response Readiness**: Regular auditing mainly helps companies in preparing for any upcoming data breach incidents. Since audits help in identifying all the weak areas, the chances for future incidents of breaches are highly reduced. This is a proactive approach that helps companies develop effective strategies for the mitigation of security breaches[43].

2.5.3. Digital signature and non-repudiation

Digital signature is value computed with a cryptographic algorithm and associated with a data object in such a way that any recipient of the data can use the signature to verify the data's origin and integrity.

Digital signatures have many applications in information security, including authentication, data integrity, and non-repudiation.

Hash function is used in conjunction with digital signature for data integrity and origin authentication : where a message is typically hashed rst, and then the hash-value, as a representative of the message, is signed in place of the original message .Digital signature o er an important advantage over MAC technique, is thenon- repudiation service. This service provide protection against false denial of involve- ment in an association [6].

2.5.4. Zero Trust

Zero Trust concept

Zero Trust is a cyber security paradigm that aims to secure resources and is based on the idea that trust must be continuously evaluated rather than given implicitly. Zero

Trust means that you should never put your trust in anyone, whether they are inside or outside the network's perimeter. It focuses on undermining trust within an organization and maintains that no implicit assumptions should be made regarding the legitimacy of individuals, devices, apps, or data that have been accessed or are being accessed on an organization's network. It offers the visibility and IT monitoring needed to keep an eye on, secure, and manage every application, user, device, and network connected to or used by the company or its employees to access data. All incoming and outgoing traffic, Traffic that stays inside the organization's boundaries is likewise seen as a potential threat[44].

Stratégie Zero Trust et Cloud computing

NIST's principles provide the foundational framework for Zero Trust architecture implementation in both on-premises and cloud environments. The strategic importance of Zero Trust in cloud environments prompted a May 2021 U.S. Executive Order for federal adoption during cloud migration, followed by CISA's June 2021 release of a reference guide and maturity model. This model (Figure 2.5) categorizes maturity into initial, advanced, and optimal levels, built upon five pillars: Identity, Hardware, Network, Application, and Data. Advancing requires stricter protective measures.

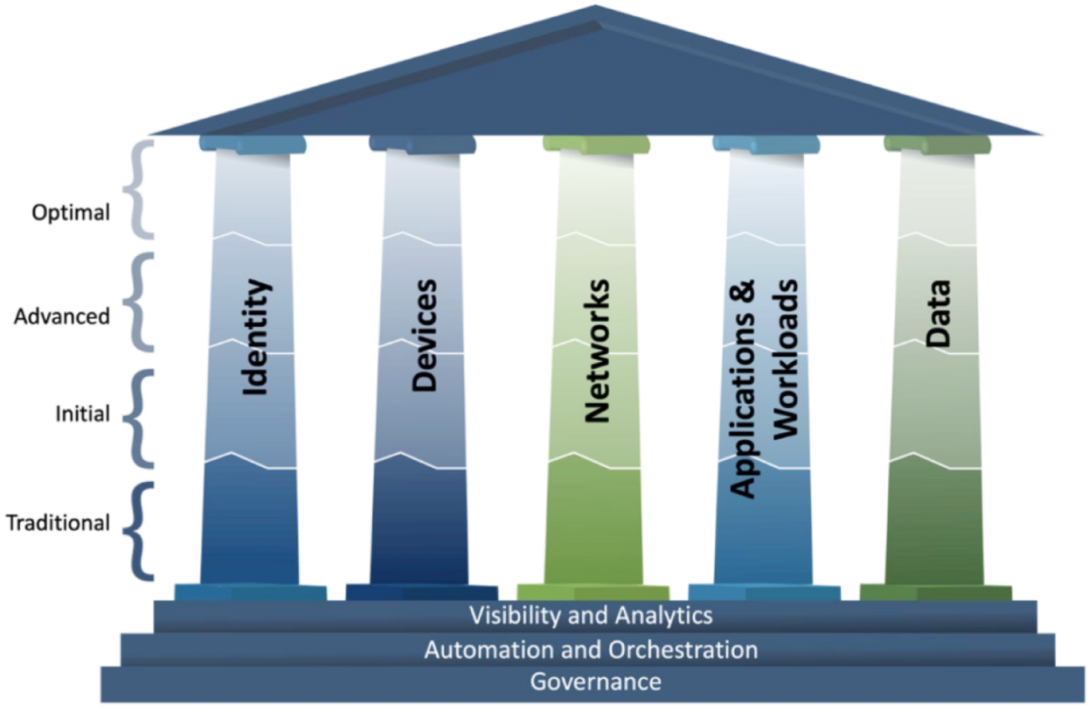


Figure2. 5: *Architecture modèle de maturité Zero Trust [45].*

The General Services Administration (GSA) guides federal agencies on Zero Trust implementation, detailing tools across key domains for tailored approaches based on capabilities, policy maturity, and related GSA offerings. Concurrently, OMB M-22-09 outlines a federal Zero Trust architecture strategy. It mandates agency cybersecurity policy compliance by FY2024, aiming to fortify U.S. IT infrastructure and citizen privacy from cyber threats.Roeser and Gottschalk, in a DoD-funded MIT Lincoln Laboratory study, defined Zero Trust as security principles treating all system entities as inherently untrusted. This architecture ensures data/application CIA via rigorous access controls, coordinated by an orchestration policy and supported by continuous monitoring and risk management. (Figure2. 6) illustrates this.

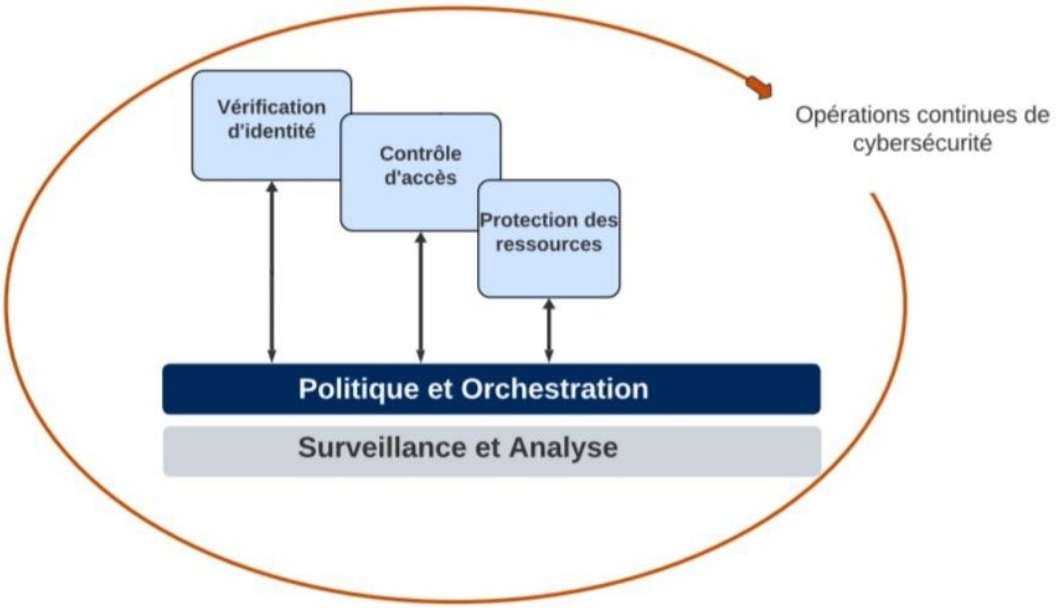


Figure2. 6:*Architecture Zero Trust Lincoln Laboratory[45].*

The studies cited above, along with the scientific and technical opinion of the National Cybersecurity Agency (ANSSI), have concluded that adopting a "Zero Trust" approach is necessary to improve the security of increasingly open, complex, and distributed IT infrastructures, such as cloud computing. However, this transition must be carried out in a gradual and controlled manner. Developing a Zero Trust architecture requires identifying the Zero Trust principles applicable to the context, as well as appropriate existing or new technical solutions (encryption, authentication, access control, etc.)[45].

3. Security Comparison Between Cloud and Blockchain

3.1. Security Dimension: Architecture & Control

- **Traditional Cloud Computing:** Centralized systems controlled by a single Cloud Service Provider (CSP). It creates a Single Point of Failure (SPoF).
- **Blockchain Technology:** Decentralized and Distributed Ledger Technology (DLT). Data is replicated across all network nodes.

3.2. Security Dimension: Data Integrity

- **Traditional Cloud Computing:** Vulnerable to alteration. Database administrators or hackers with high privileges can modify or delete historical records.
- **Blockchain Technology:** Fully Immutable. Once data is verified through consensus algorithms, cryptographic hashing prevents any retroactive tampering.

3.3. Security Dimension: System Availability

- **Traditional Cloud Computing:** Highly vulnerable to downtime if central data centers suffer severe hardware failures or targeted DDoS attacks.
- **Blockchain Technology:** Extremely high fault tolerance.

The infrastructure remains functional and online even if multiple individual nodes fail.

3.4. Security Dimension: Trust Model

- **Traditional Cloud Computing:** Requires full reliance on a Trusted Third Party to protect, store, and manage user data privacy.
- **Blockchain Technology:** Operates on a "Trustless" architecture. Security and operational validation are enforced mathematically via Consensus Protocols.

3.5. Security Dimension: Access Control & Encryption

- **Traditional Cloud Computing:** Relies heavily on traditional network perimeters such as firewallls and Identity and Access Management (IAM) systems.
- **Blockchain Technology:** Secured natively through asymmetric cryptography (Public and Private Key pairs) and managed automatically via Smart Contracts [46].

4. Comparative Analysis of Cloud and Blockchain Security

FEATURE	CLOUD COMPUTING	BLOCKCHAIN TECHNOLOGY
Data Storage	**Centralized:** Data Is Stored On Servers Owned By Cloud Providers.	**Decentralized:** Data Is Replicated And Distributed Across A Network Of Computers.
Data Immutability	**Changeable:** Data Can Be Modified Or Deleted With Proper Authorization.	**Unchangeable:** Once Recorded, Data Cannot Be Altered Or Deleted.
Security	**Third-Party Reliance:** Security Depends On The Cloud Provider's Measures.	**Cryptographic Security:** Uses Cryptography And Distributed Ledger For Tamper-Proof Data.
Scalability	**High Scalability:** Resources Can Be Easily Scaled Up Or Down On Demand.	**Potential Limitations:** Transaction Processing Speed Can Be Slower On Some Blockchains.
Cost	**Variable Costs:** Pay-As-You-Go Model For Resources Used.	**Potentially Higher Costs:** Transaction Fees And Energy Consumption Can Be Higher For Some Blockchains.

Table1. 2: Comparative Analysis of Cloud and Blockchain Security[47].

5. INTEGRATION OF CLOUD AND BLOCKCHAIN

5.1. BLOCKCHAIN SUPPORT FOR CLOUD COMPUTING

Blockchain integration with cloud computing brings us into the next era of data security and service availability. Blockchain overcomes most of the research issues of the cloud with its characteristics.

5.1.1. INTEROPERABILITY

In public clouds, internal communication is not allowed, and it makes many industries back off from using the cloud. When cloud integrated with blockchain,

consider the different clouds as nodes. Inter-node communication is possible in the blockchain. All the nodes present in the same network share the data among themselves so that every node contains a copy of transactions. It brings us transparency into the network. They update every next transaction into the ledger, which publishes to all other nodes. In this way, companies can add any number of networks and can preserve the accessibility of the data, which brings authenticity into the network.

5.1.2. DATA ENCRYPTION

We all know the data is decrypted before storing it in the cloud, which questions the data integrity. In the blockchain network, all the block data is turned into a hash code using cryptographic algorithms, and it generates a hash key for each block. Let us consider a scenario in which blockchain is used to preserve task scheduling in the cloud. To ensure timeliness and permanent data integrity, the control system that collects data from the task scheduling produces hash code and records it in the blockchain network immediately. Because the blockchain has the facility for block discovery consensus mechanisms, block data integrity is maintained. Each node in the network contains a copy of each transaction that provides us with the availability and persistence that helps the network withstand potential fault points and attacks. While cloud-collected data is reliable, the blockchain nodes maximize data availability and data validity by projecting it as an on-demand service with no downtime.

5.1.3. SERVICE LEVEL AGREEMENTS

These agreements in the cloud are favorable to the service provider or customer without equal justice. To solve this issue, we can make use of blockchain smart contracts [48].

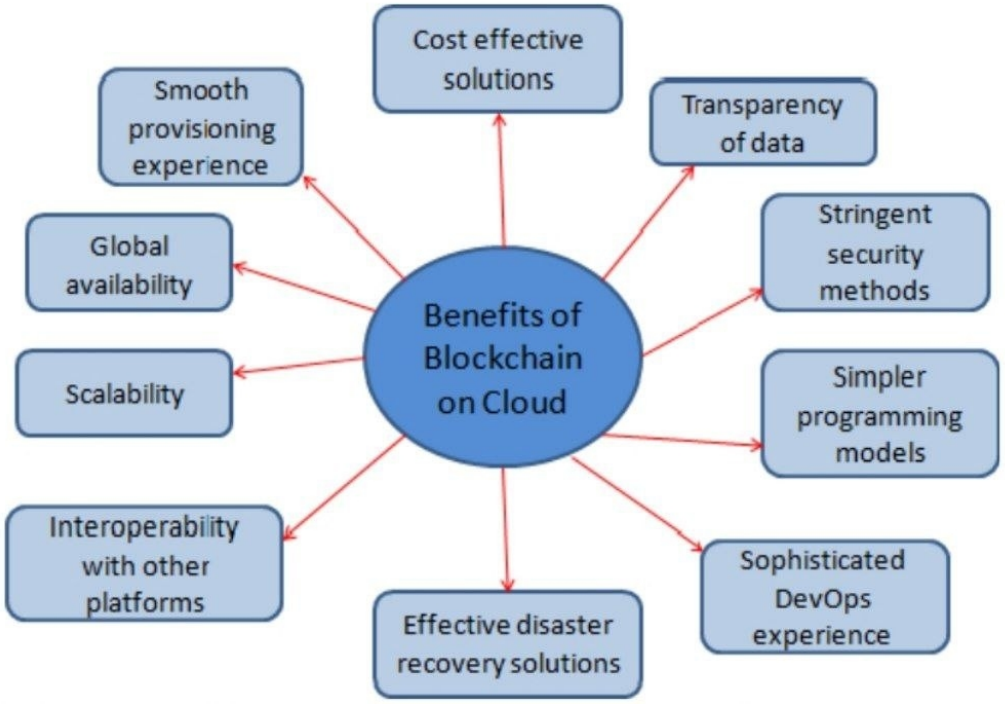


Figure2. 7: *Benefits that Blockchain providing to cloud [48].*

Conclusion

In this chapter, we explored the fundamental concepts of Cloud Computing, its architecture, models, advantages, and security mechanisms. We also examined the main security challenges associated with cloud environments. Furthermore, a comparison between Cloud Computing and Blockchain was presented, highlighting their differences and complementary features. This analysis provides a foundation for understanding how these technologies can be used individually or combined to build more secure, efficient, and reliable systems.

Application Design and UML Modeling

Chapitre 03

Chapitre03: Application Design and UML Modeling

Introduction

the previous chapter, we presented the concepts of Blockchain and Cloud Computing, as well as their main characteristics and advantages. We also explained the relationship between these two technologies and their role in improving data security, reliability, and management. In this chapter, we move to the design phase of our application using UML (Unified Modeling Language). This phase allows us to model the different components of the system and represent the interactions between actors and application functionalities through various UML diagrams.

1. What is UML?

The Unified Modeling Language (UML) is a standard modeling language used to visualize, specify, construct, and document software systems. It provides developers with graphical tools that help in understanding, designing, and organizing the different components of a system. UML is widely used in software engineering because it simplifies the representation of system structure and behavior through a set of standardized diagrams. It also facilitates communication between developers, analysts, and users during the development process. Over time, UML has evolved through several versions in order to support new technologies and improve software modeling techniques. Today, UML is considered one of the most important modeling

tools for system analysis and design. In our project, UML is used to model the application and represent the interactions between the different actors and system components using various diagrams such as use case diagrams, class diagrams, sequence diagrams, and activity diagrams [49].

2. Unified Modeling Language (UML) Diagrams

Among the most commonly used UML diagrams are the Use Case Diagram, Sequence Diagram, and Activity Diagram.

● Use Case Diagram

A Use Case Diagram is a UML diagram that represents the functionalities provided by a system and the interactions between the system and its users (actors). It helps identify system requirements from the user's perspective and provides a high-level view of the system's behavior.

● Sequence Diagram

A Sequence Diagram is a behavioral UML diagram that illustrates how objects interact with each other through messages arranged in a chronological order. It is used to describe the flow of a specific scenario and the sequence of operations required to accomplish a task.

● Activity Diagram

An Activity Diagram is a behavioral UML diagram used to model the workflow of activities and actions within a system. It shows the flow of control from one activity to another, including decisions, parallel processes, and the overall execution path of a process[50].

3. SYSTEM REQUIREMENTS ANALYSIS

This section establishes the technical specifications and requirements engineering for the Hybrid Cloud-Blockchain Student Management System. It comprehensively covers the system actors, specific functional operations, and non-functional quality constraints necessary to ensure reliable performance and strict data integrity.

3.1. Identification of Acteurs

Based on the architectural scope and Use Case definitions of the system, a primary academic actor is identified:

- **Academic Officer (User):** The authorized personnel responsible for academic data entry, auditing, verification, and batch processing. This actor possesses the necessary credentials to interact with both the cloud collection and the decentralized ledger.

3.2. Functional Requirements

Functional requirements specify the core algorithmic capabilities and operational workflows that the software application must perform. The system functions are cataloged below:

Functionality / Feature	**Description**
User Navigation Dashboard	Enables the user to navigate directly between registration, lookup, and auditing interfaces via a unified control panel.
Single Student Onboarding	Allows the entry of individual student records (Name, Student ID, Academic Level) and securely commits them to the system storage.
Mass Data Ingestion	Uploads and parses external text or CSV files containing large-scale student datasets to process them simultaneously in bulk.
Performance Benchmarking & Data Display	Tracks and calculates operational throughput and network latency in milliseconds (ms), and renders the managed records and measured metrics directly on the screen upon execution.
Cloud Asset Management	Retrieves, displays, updates, and deletes active student profiles directly from the connected database layer.

Cryptographic Ledger Auditing	Queries and displays immutable, digitally signed student transactions and block structures within the decentralized ledger.
Automated Fallback Verification	Executes record verification via Student ID on the blockchain network, with an automatic redirection to the cloud layer if node latency or network interruption occurs.

Table3. 1: Functional Requirements

3.3. Non-Functional Requirements

Non-functional requirements describe behavioral constraints, environmental boundaries, and structural quality attributes imposed on the application:

- **Immutability & Tamper-Evidence:** All academic certificates and data streams anchored onto the BigchainDB blockchain network must be digitally signed via EdDSA cryptographic keypairs, preventing unauthorized alterations or post-facto deletion on the ledger.
- **High Availability & Fallback Resilience:** To ensure business continuity during network partitions or blockchain node timeouts, the software must deploy an automated failover mechanism (Fallback) that routes read queries immediately to the local MongoDB Atlas backup database, eliminating UI rendering failures.
- **UI Ergonomics & Component Styling:** The graphic interface must follow modern design standards using unified style aesthetics (FlatLaf Look & Feel), distinct brand color maps (Cloud: Dark Blue, Blockchain: Amber/Orange), and strict client-side data pattern validation.

3. System Diagrams Documentation

3.1. Use Case Diagram

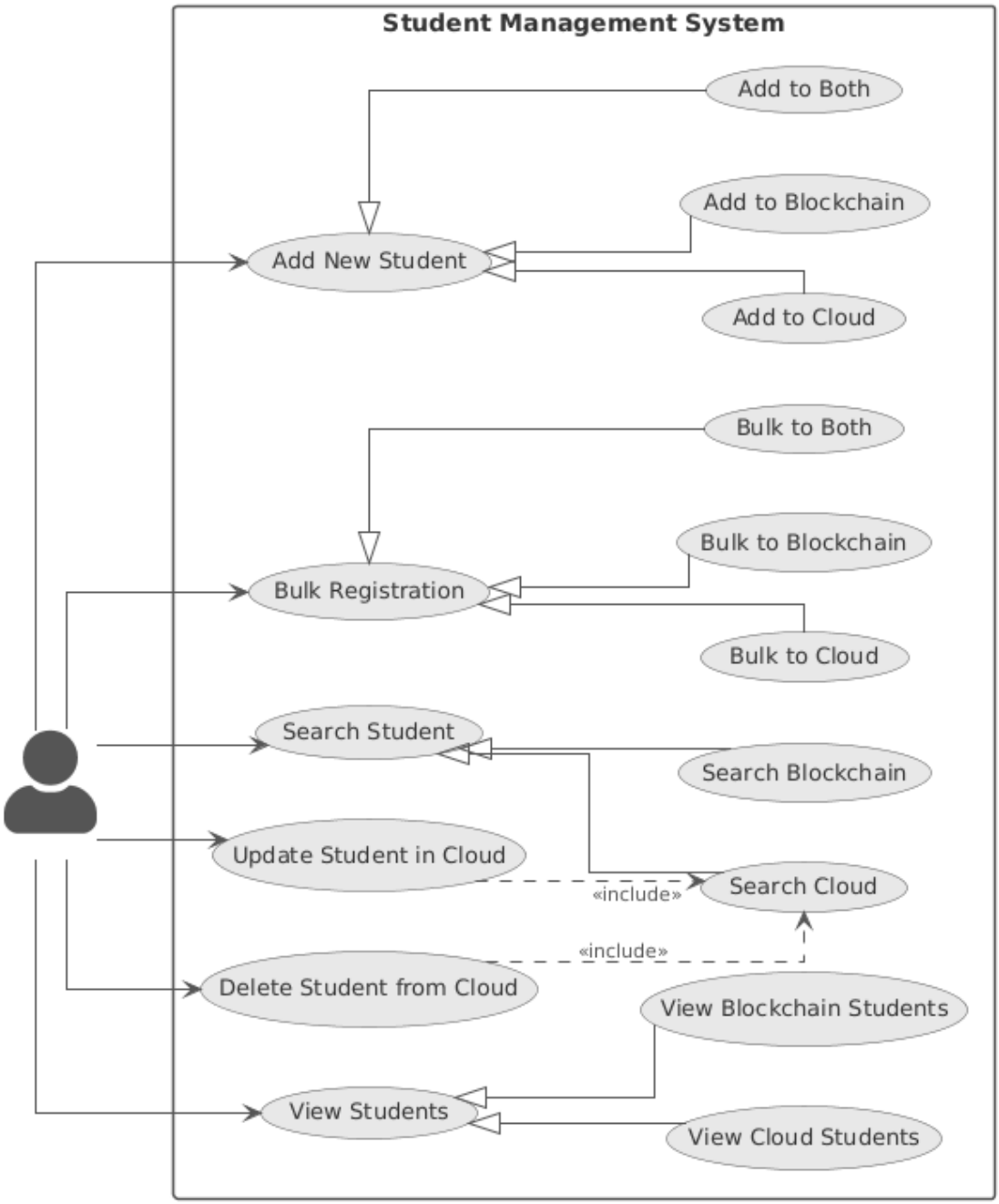


Figure3.1: Use Case Diagram

3.2. Activity Diagrams

3.2.1. Add New Student to Both(Cloud+Blockchain)

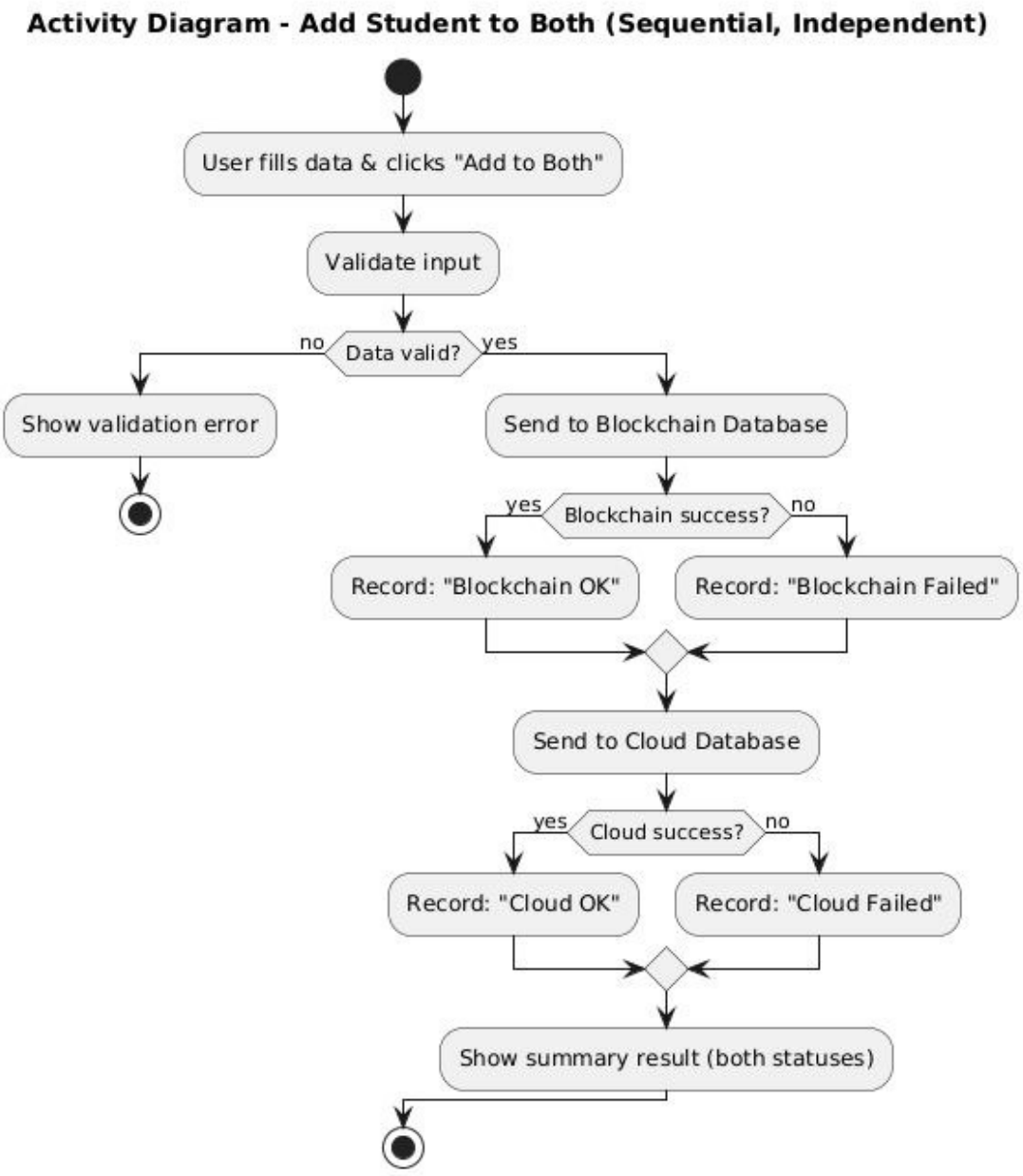


Figure3. 2 : Activity Diagram_ Add New Student to Both(Cloud+Blockchain)

3.2.2. Add New Student to Cloud Only

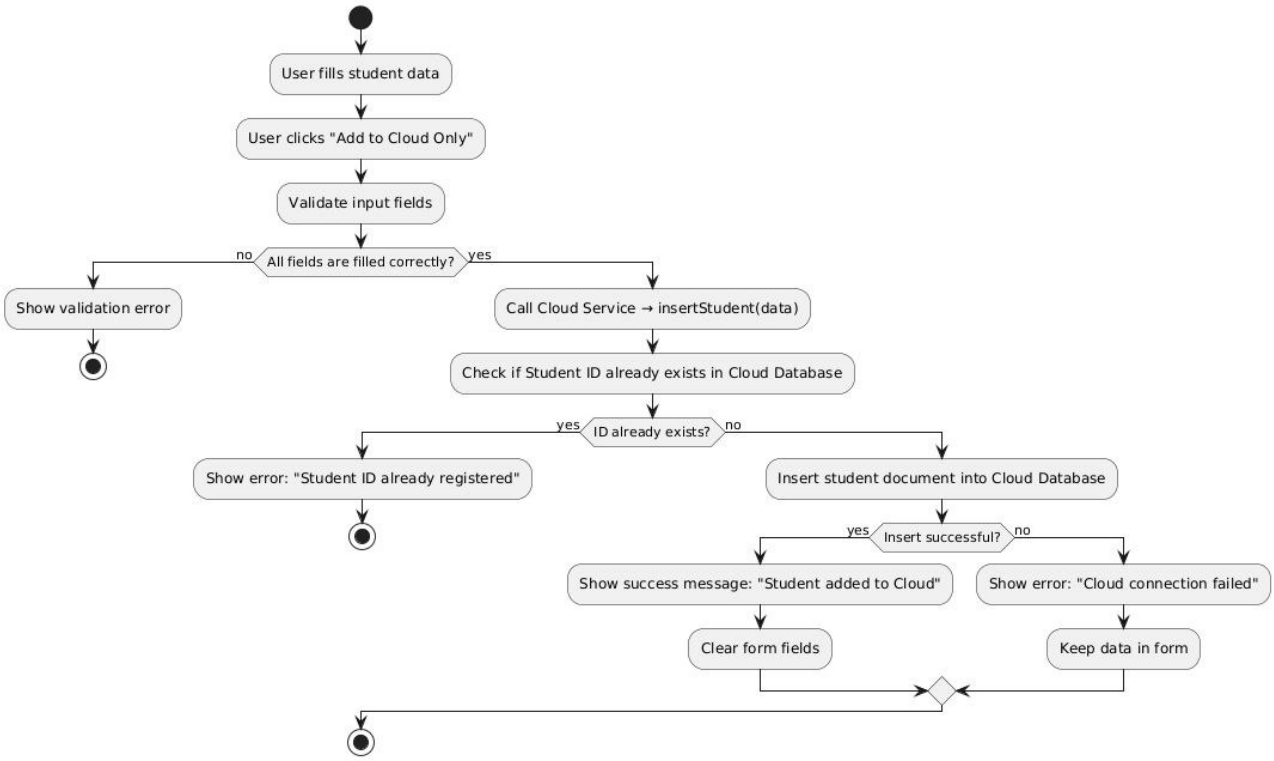


Figure3.3 : Activity Diagram_Add New Student to Cloud Only

3.2.3. Add Student to Blockchain Only

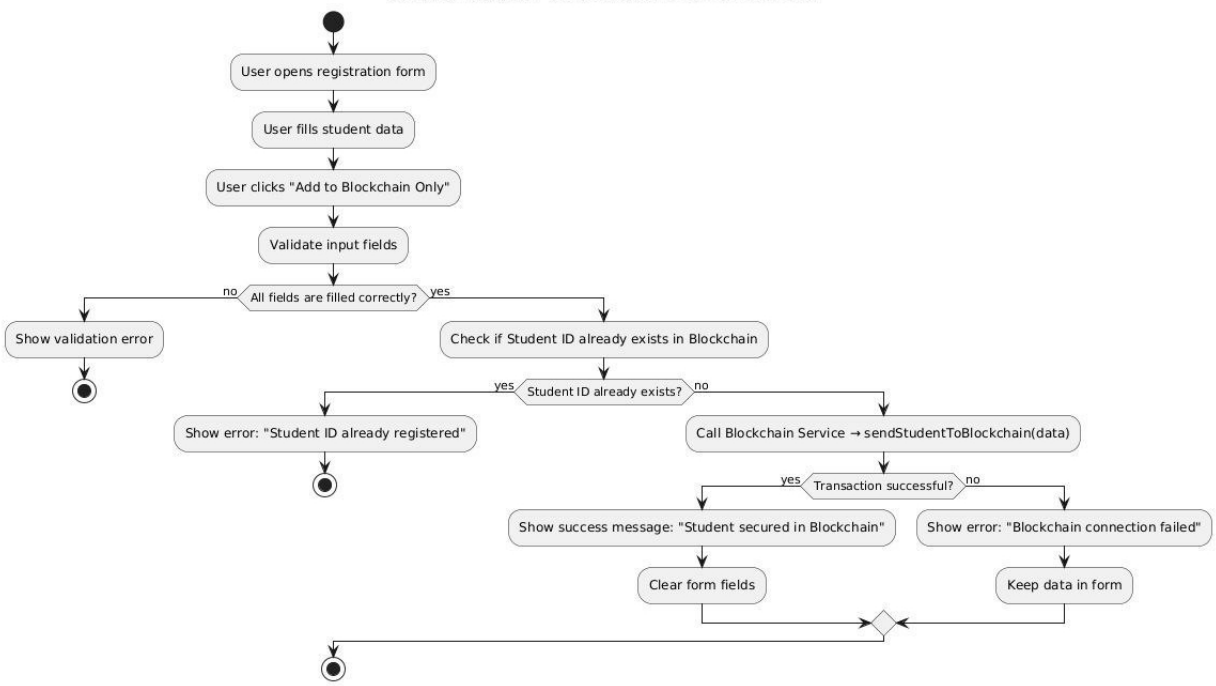


Figure3. 4: Activity Diagram_Add Student to Blockchain Only

3.2.4. Search Student in Cloud

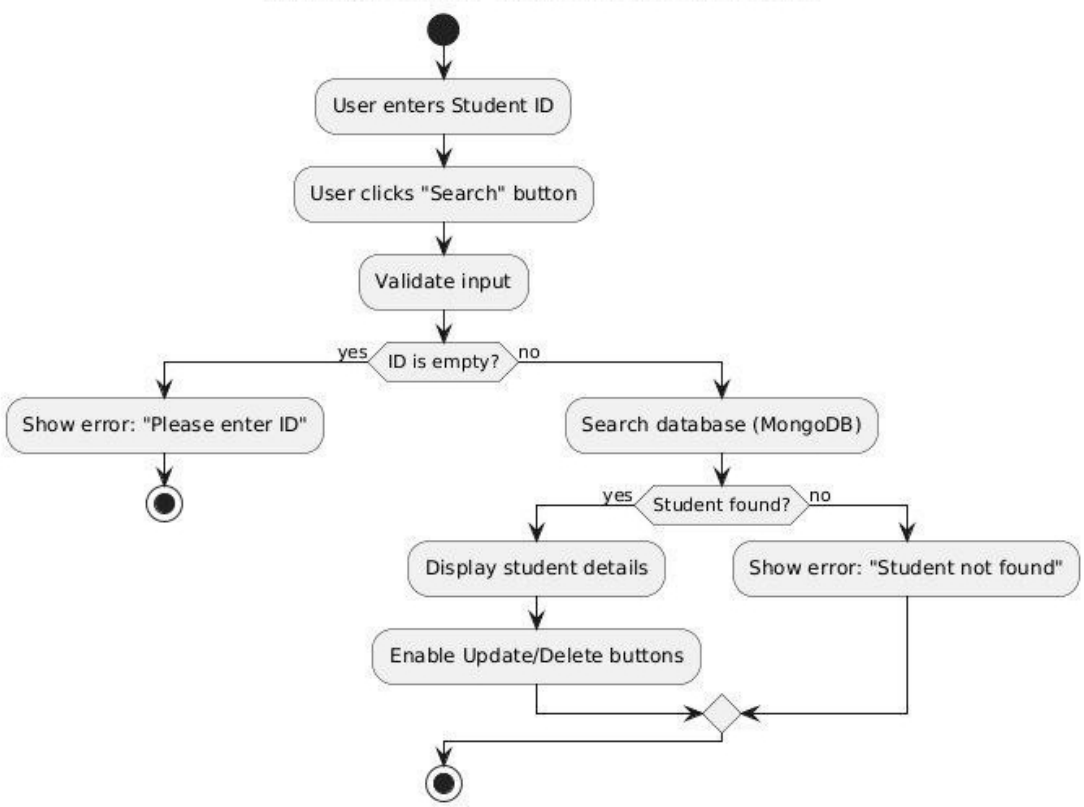


Figure3.5:Activity Diagram_Search Student in Cloud

3.2.5. Search Student in Blockchain

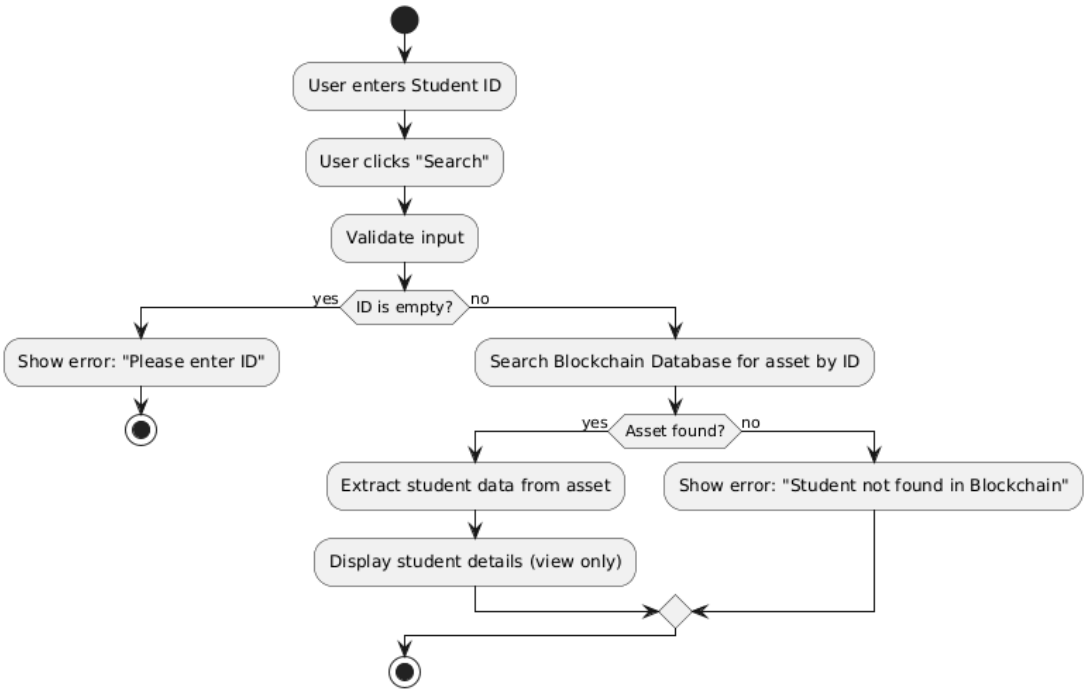


Figure3.6: Activity Diagram_Search Student in Blockchain

3.2.6. Get All Students from Cloud

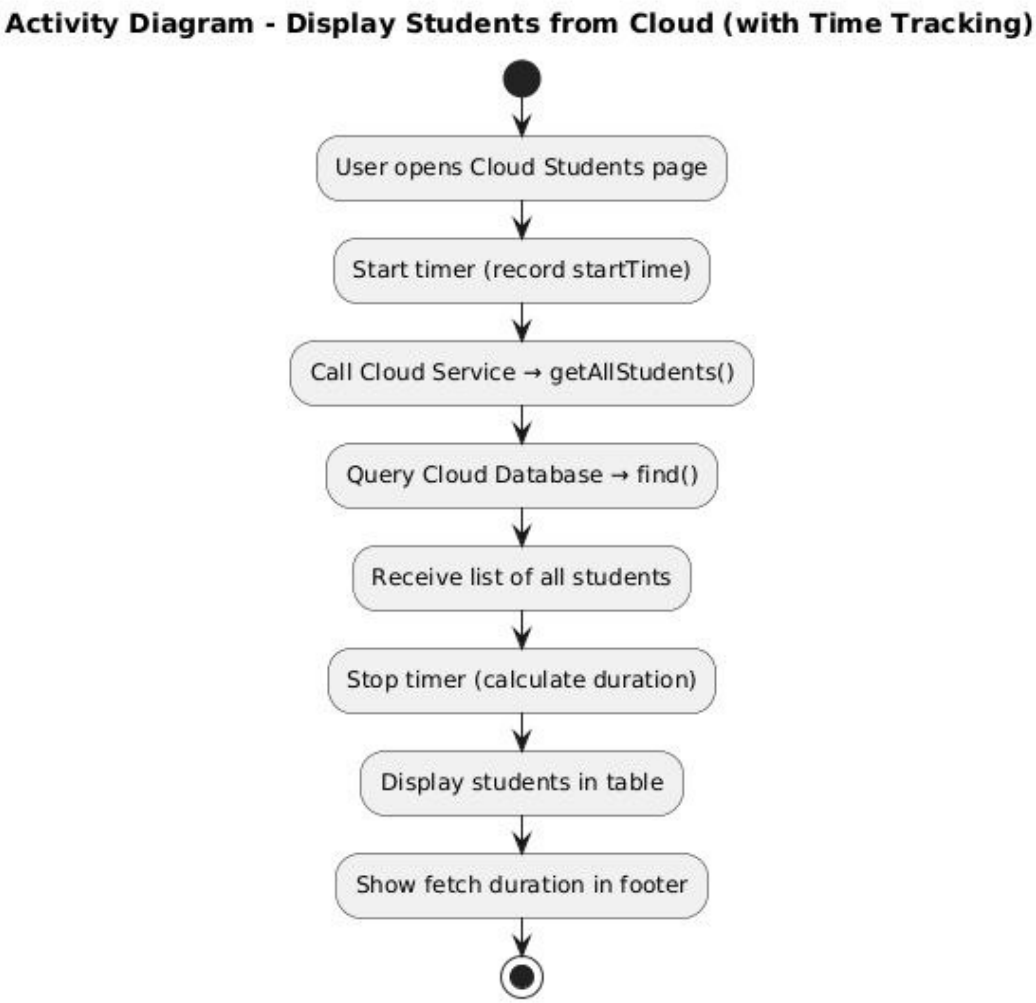


Figure3.7: Activity Diagram_Get All Students from Cloud

3.2.7. Get All Students from Blockchain

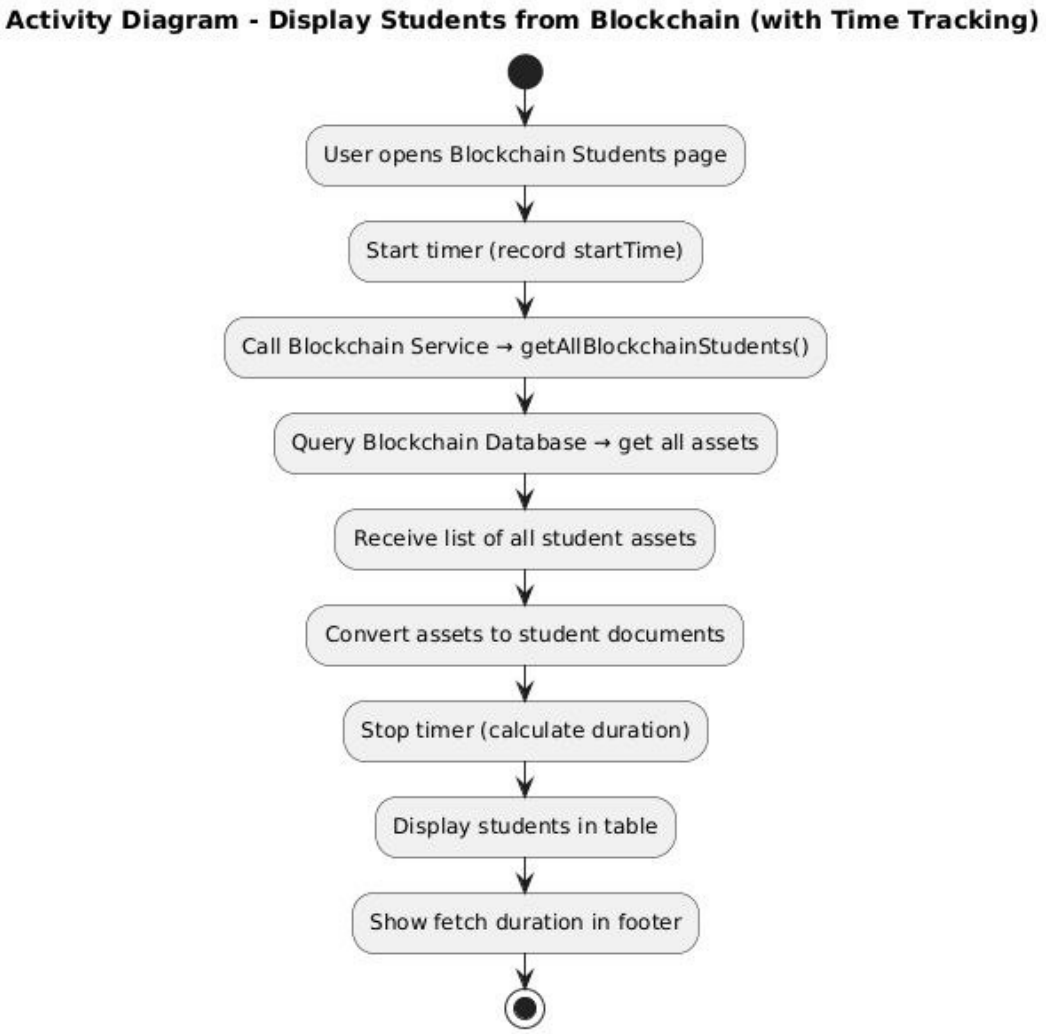


Figure3. 8: Activity Diagram_Get All Students from Blockchain

3.2.8. Bulk Registration

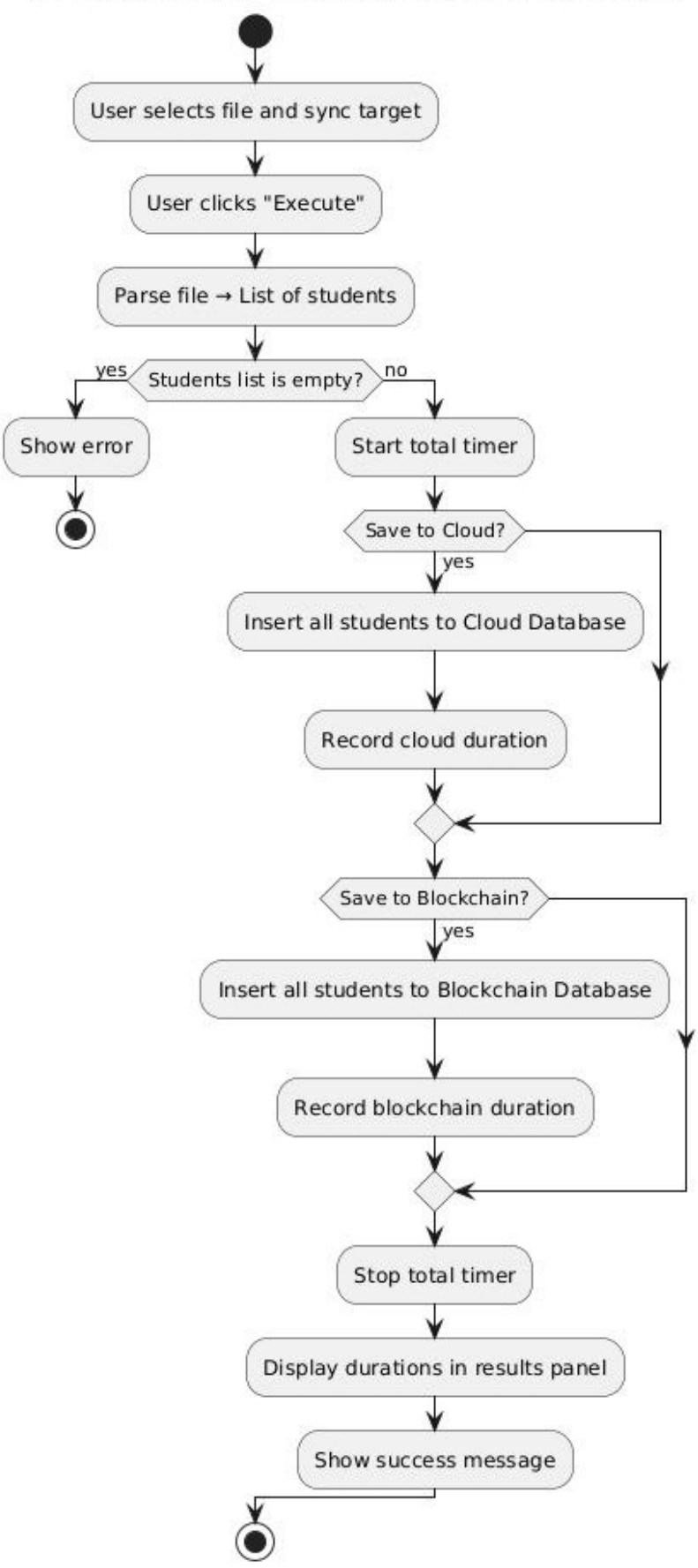


*Figure3.9:*Activity Diagram_ *Bulk Registration*

3.3. Sequence Diagrams

3.3.1. Add Student to Both (Cloud + Blockchain)

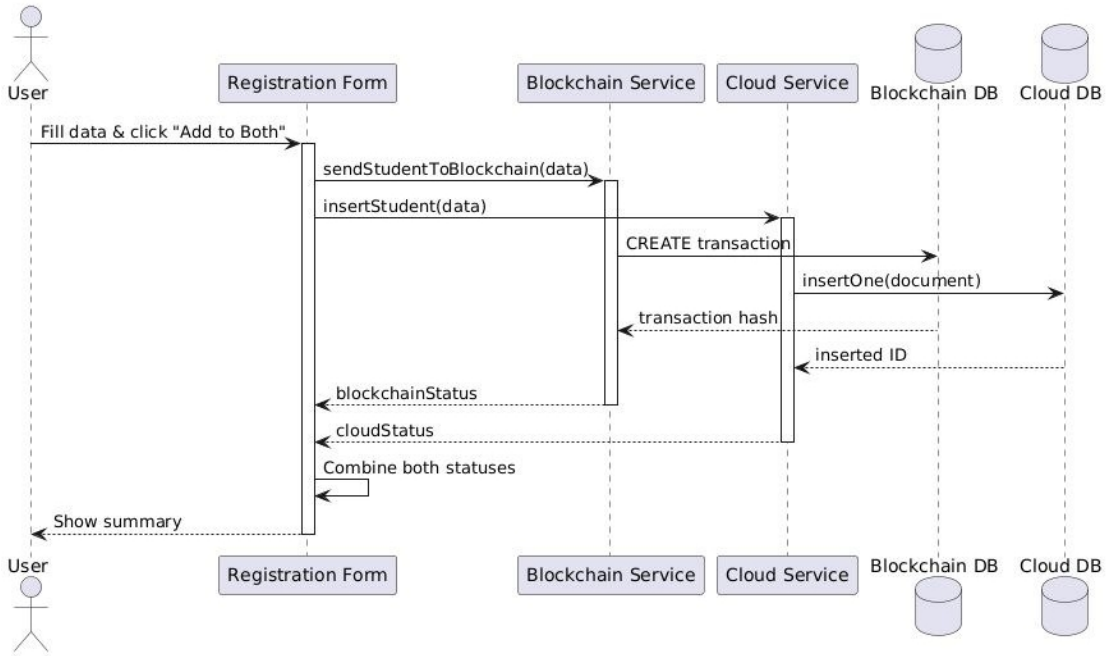


Figure3.10: Sequence Diagrams Add Student to Both (Cloud + Blockchain)

3.3.2. Add New Student to Cloud Only

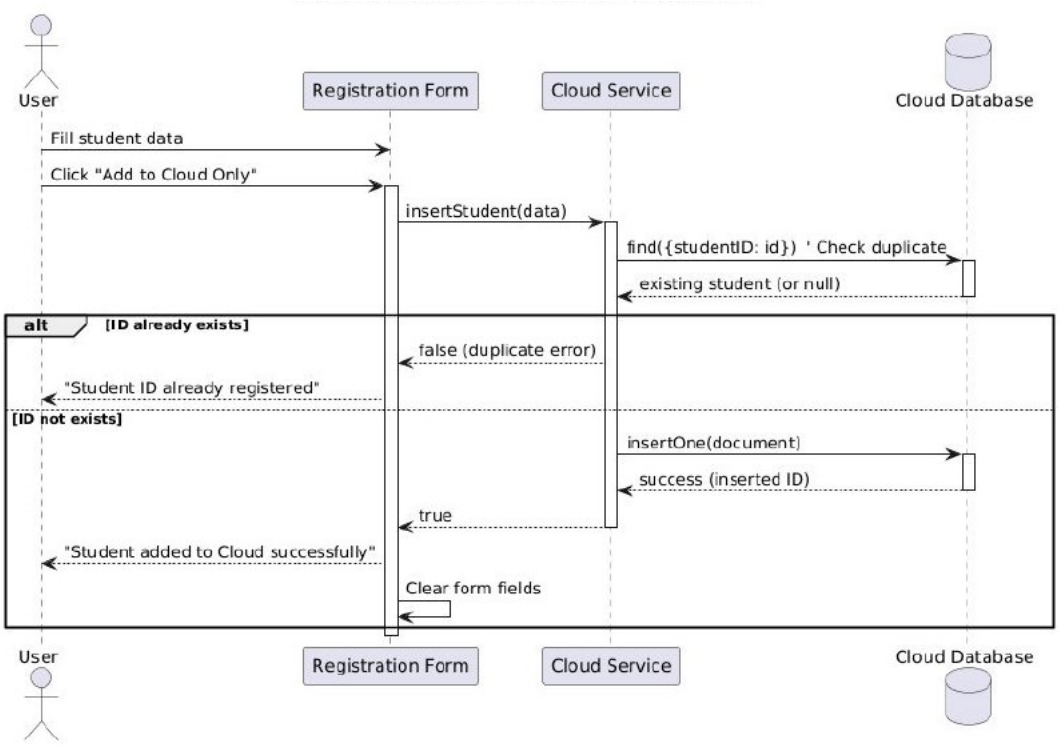


*Figure3. 11: Sequence Diagrams*_Add New Student to Cloud Only

3.3.3. Add Student to Blockchain

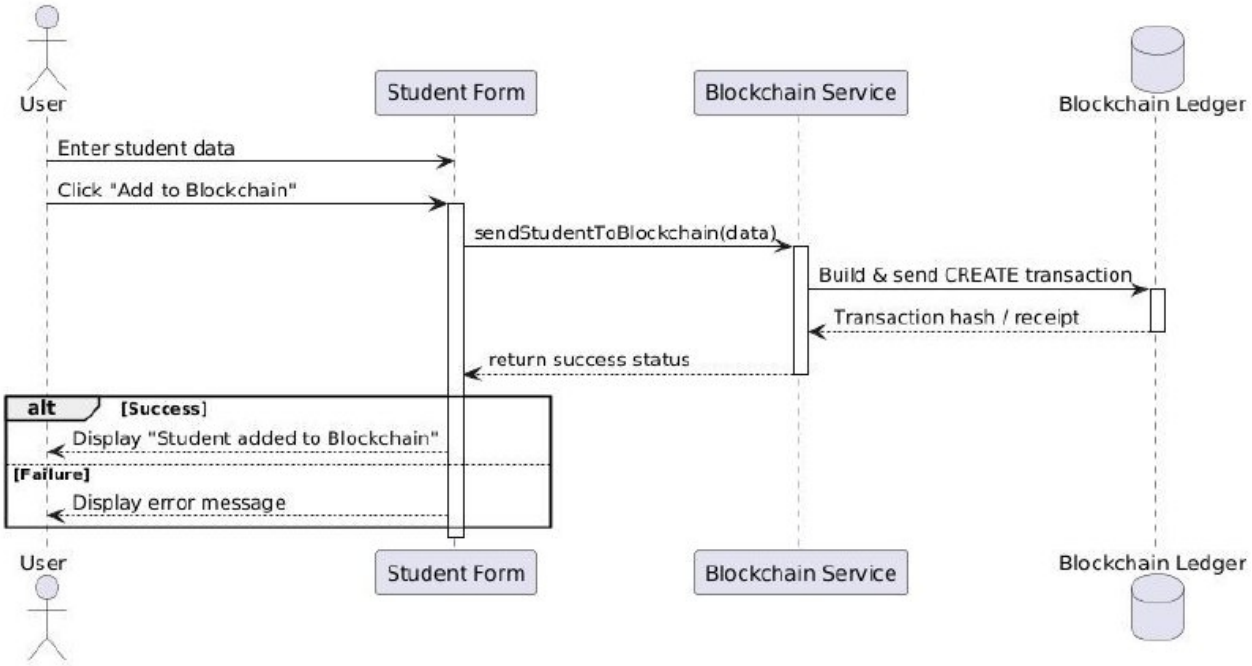


Figure3.12 Sequence Diagrams Add Student to Blockchain

3.3.4. Search Student in Cloud

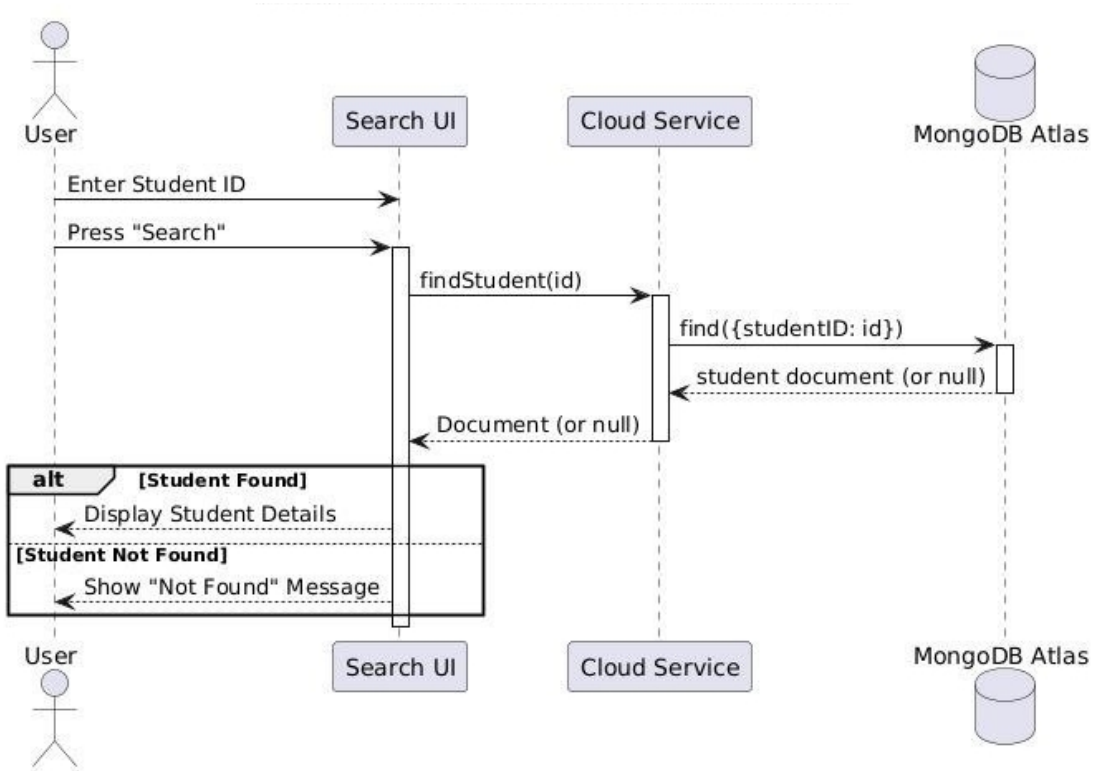


Figure3. 13: Sequence Diagrams_ Seach Student in Cloud

3.3.5. Search Student in Blockchain

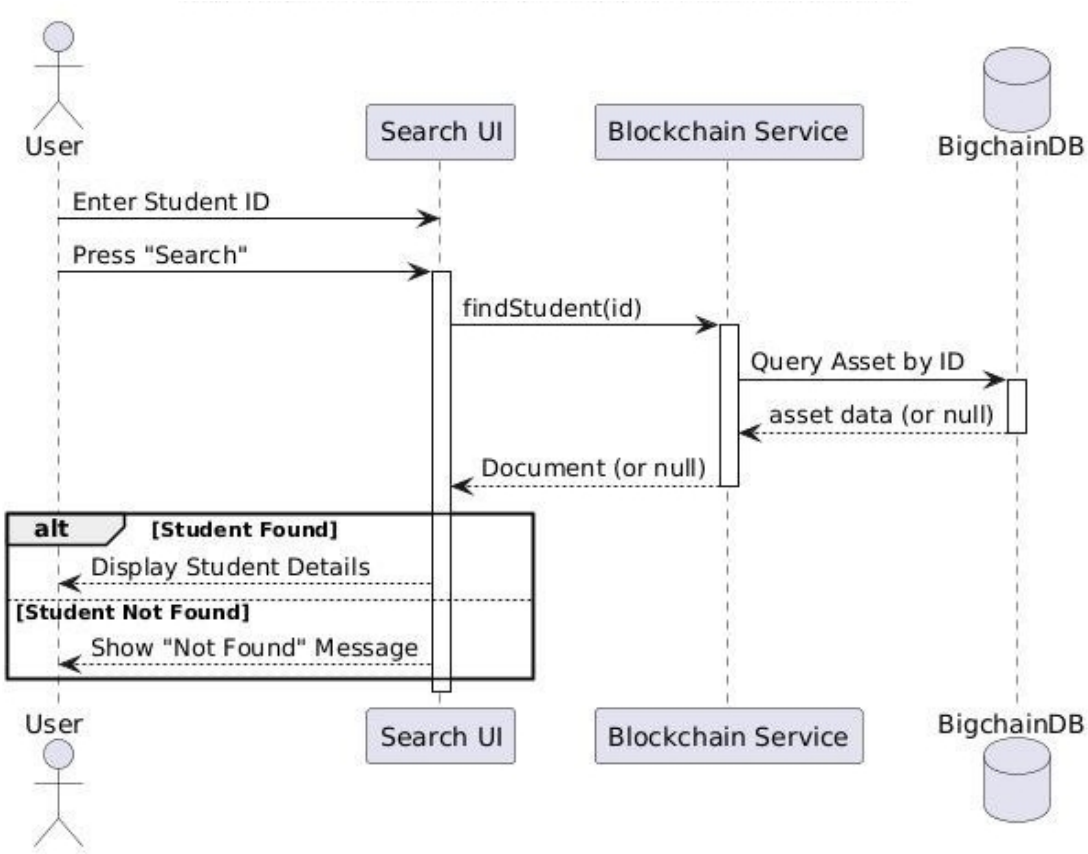


Figure3. 14 : Sequence Diagram_Search Student in Blockchain

3.3.6. Get All Students from cloud

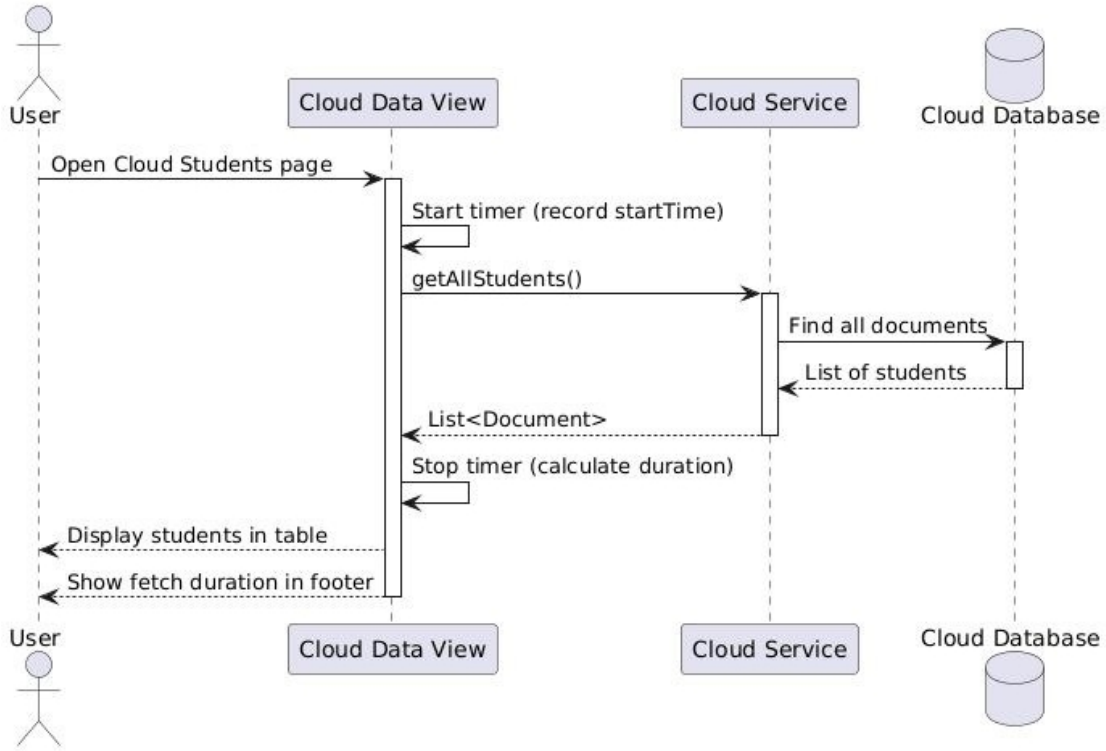


Figure3. 15 : Sequence Diagram_Get All Students from cloud

3.3.7. Get All Students from Blockchain

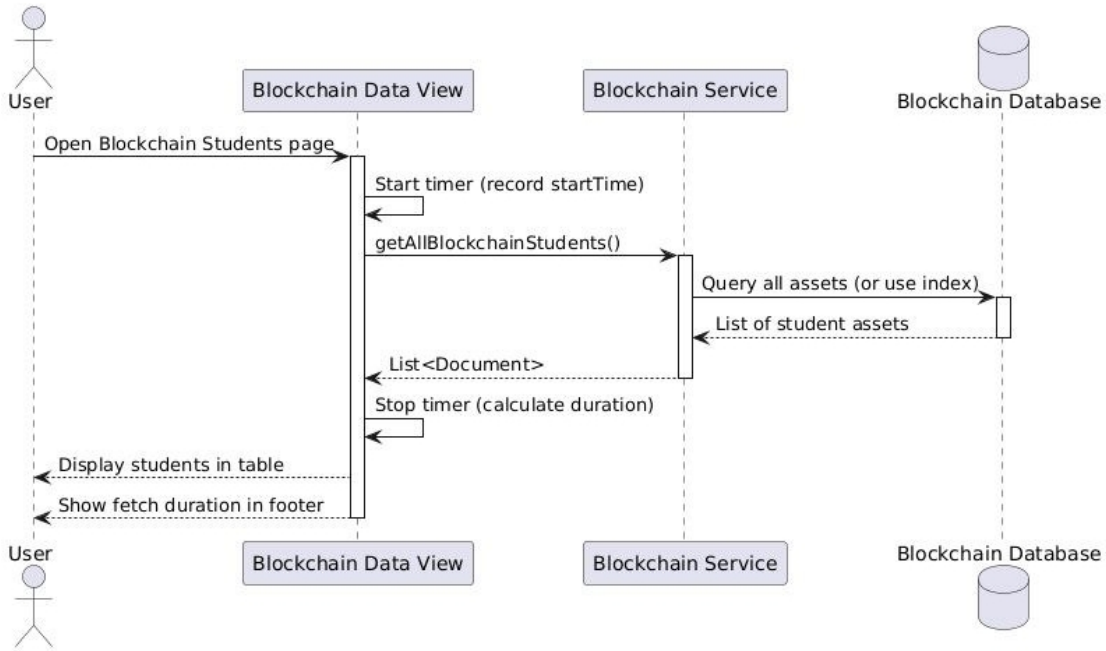


Figure3. 16 : Sequence Diagram_Get All Students from Blockchain

3.3.8. Bulk Registration

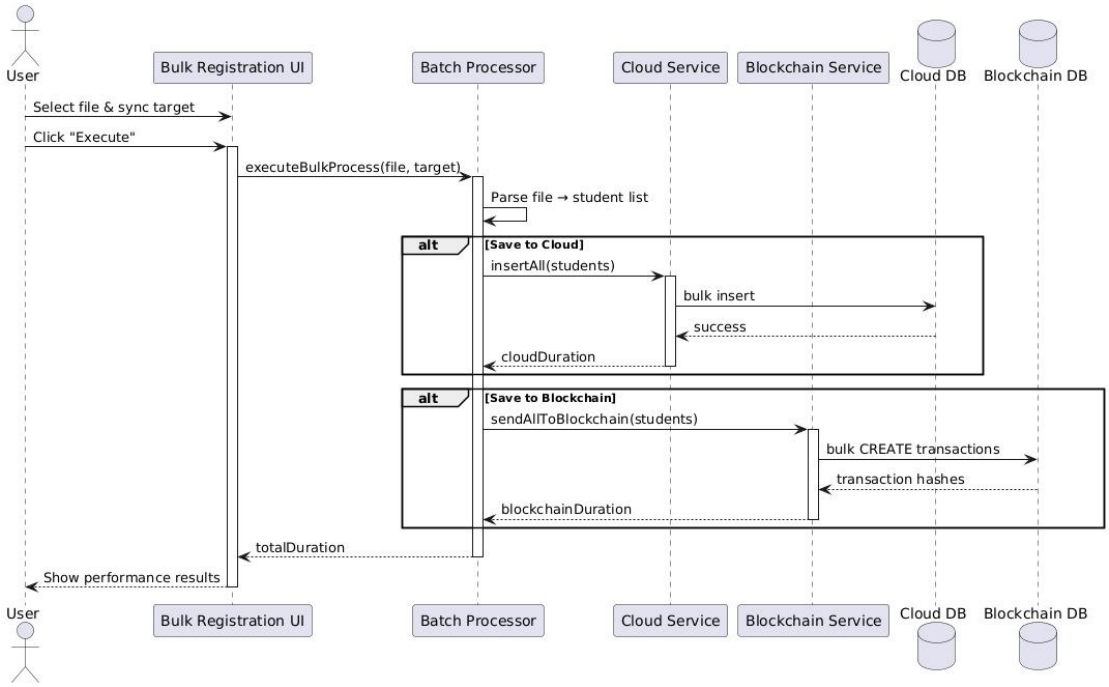


Figure3. 17: Sequence Diagram_ Bulk Registration

Conclusion

In conclusion, this chapter has successfully finalized the design and theoretical framework of the proposed system. We analyzed the requirements and modeled the system's architecture through structured UML diagrams, defining user interactions. With this architectural blueprint complete, the groundwork is now laid to transition into the fourth and final chapter, which will focus on the practical implementation, the development environment, and the empirical performance comparison between cloud and blockchain storage.

Implementation and Performance Evaluation

Chapitre 04

Chapitre04: System Implementation and Performance Evaluation

Introduction

After constructing and designing the Unified Modeling Language (UML) diagrams in the previous chapter, this fourth and final chapter transitions into the practical implementation of the research. Here, we present the development environment, the programming tools utilized to build the application, and the graphical user interfaces (GUIs) designed to manage data storage. Furthermore, this chapter primarily focuses on conducting empirical benchmarking and performance testing to evaluate and compare cloud storage against blockchain storage in terms of response time and data processing efficiency, thereby highlighting the fundamental trade-offs between both architectures.

1. utils de développement et langages utilisés

Premièrement, les principaux outils utilisés lors de la construction de notre système sont les suivants:

1.1. Eclipse



Figure4. 1*: logo Eclipse*

Eclipse is the first Java IDE. Developed by IBM using its ancestors' Age and Visual Age for Java. Since then, it has been made open source, and the Eclipse Foundation continues to oversee Its evolution. Eclipse's uniqueness stems from its modular architecture, which is entirely based on the concept of plugins. All of this software's features are designed as plugins. Eclipse's license allows for the provision of both open-source and closed-source plugins, as well as free or paid plugins. This is the reason it is now more than just a Java IDE and supports a wide variety of programming languages [51].

1.2. Java Environment



Figure4. 2: logo java

Java is both a programming language and a software environment designed to operate consistently across different platforms. It also provides a rich set of APIs that support software development. Java is a programming language that combines several important concepts, including object-oriented programming, strong typing, security mechanisms, multithreading, and exception handling. Java also represents an execution environment based on virtual machine t9999echnology, developed over many years to ensure code portability while maintaining good performance. In addition, Java includes a wide range of APIs that facilitate the development of

information systems, such as database management, text processing, audio and image handling, networking, and distributed architectures. This section is divided into three parts. The first part introduces Java as a programming language. The second part explains Java as an abstraction of a homogeneous execution environment. The third part presents Java as an environment that integrates useful APIs for developing information systems, especially systems related to mobility and application persistence [52].

1.3. MongoDB Atlas



Figure4 3 : MongoDB Atlas

MongoDB Atlas is a multi-cloud database service (database as a service) created by the same company that developed MongoDB. MongoDB Atlas nodes can be accessed and deployed across other cloud providers such as Microsoft Azure, AWS, and GCP. MongoDB Atlas offers the ability to create fully managed global clusters. These clusters can be scaled horizontally and vertically based on workload and storage requirements. MongoDB Atlas clusters come with a range of additional settings that can be configured according to client preferences [53].

1.4. BigchainDB



Figure4. 4: BigchainDB

is an open-source distributed database that combines traditional database features with

blockchain characteristics such as decentralization, immutability, high throughput, and secure data storage. It supports multiple programming languages including Java, Python, and JavaScript, and allows easy deployment on virtual machines. BigchainDB mainly uses two types of transactions: CREATE transactions for creating new records or assets, and TRANSFER transactions for transferring the ownership or control of existing assets [54].

2. System Implementation and Performance Evaluation

This section presents the empirical implementation of the Hybrid Cloud-Blockchain Student Management System. It outlines the graphical user interfaces (GUIs) developed to handle academic operations, presenting them in their precise operational execution sequence.

2.1. User Interfaces Description

2.1.1. MainApplication

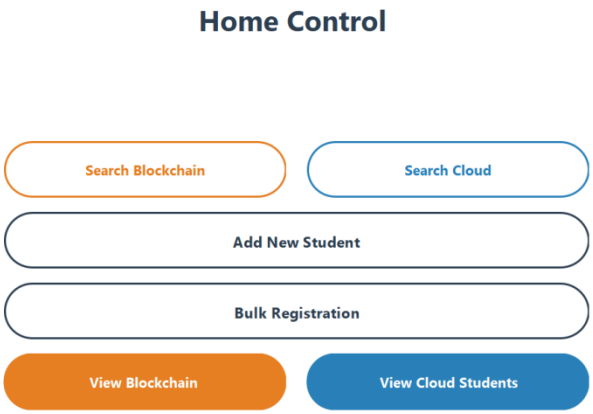


Figure4. 5 : Main Application Dashboard (HomeForm)

This interface serves as the central control panel of the system, enabling the user to navigate instantly and seamlessly between the different operational environments available in the application. The interface is designed to provide direct access to core operations through a structured set of interactive buttons.

2.1.2. Single Student Registration Form

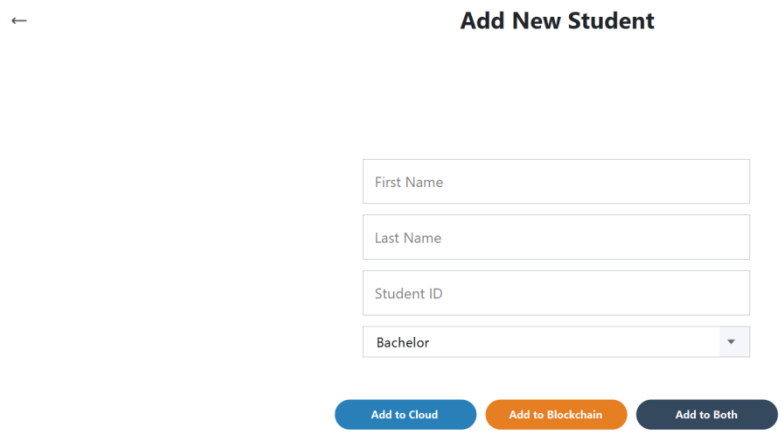


Figure4. 6: Individual Student Registration Interface

This interface is used to enter individual student records into the system. It contains specific input fields for: First Name, Last Name, Student ID, along with a drop-down menu to select the academic level (Bachelor, Master, PhD). When submitted, the data is passed to the backend to be saved across the connected storage layers.

2.1.3. Bulk Student Registration Module

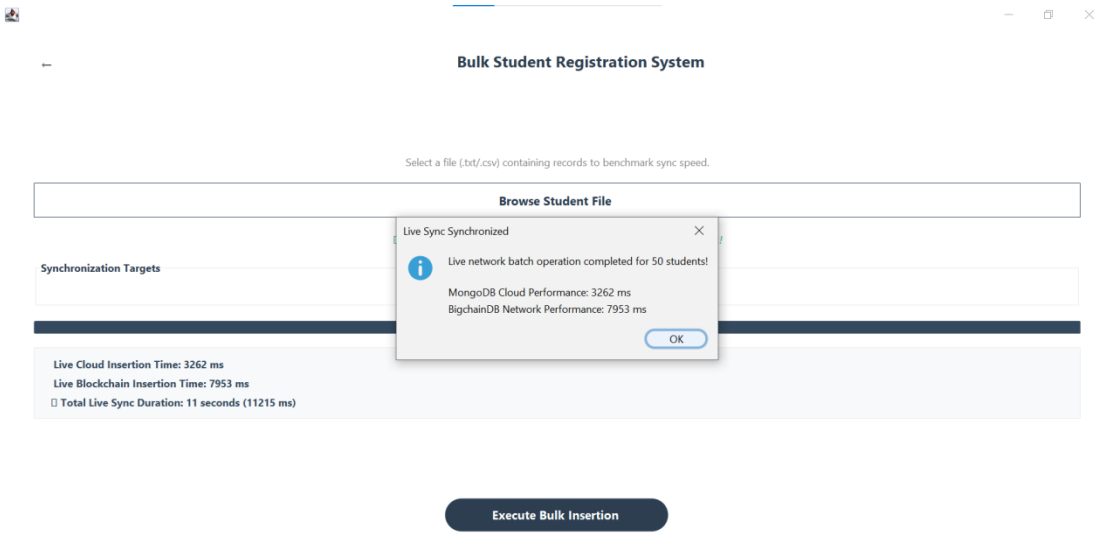


Figure4. 7: Automated Bulk Registration and Benchmark Interface (BulkRegistrationForm)

This screen allows users to upload and import large-scale student data from text or CSV files. The interface includes a browsing tool to select files from the local device, and radio buttons that let the user choose the storage path: either sending the data stream to both the cloud and blockchain environments simultaneously, or routing it to a single infrastructure layer. A progress bar displays the real-time status of the operation, and this screen is used to execute volume stress tests and capture performance metrics.

2.1.4. Cloud Data Viewer

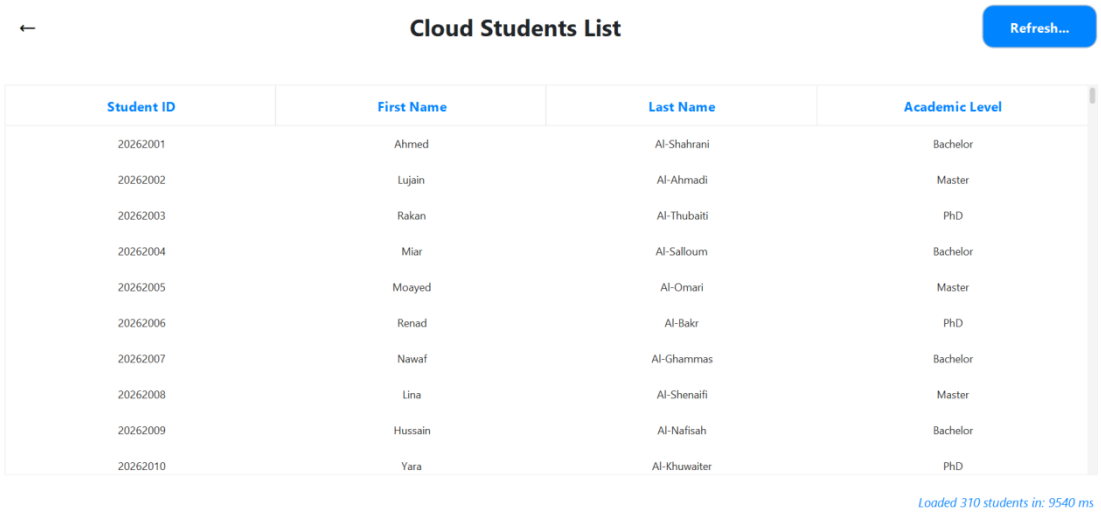


Figure4. 8 : Synchronized Cloud Storage Data Viewer (DisplayCloudyDataForm)

This screen displays student records stored in the cloud database inside a scrollable table view. The data is loaded in the background to prevent the user interface from freezing while fetching records from the remote server. At the bottom, the screen includes a digital counter that calculates and displays the exact data retrieval and display speed in milliseconds (ms).

2.1.5. Cloud Data Search

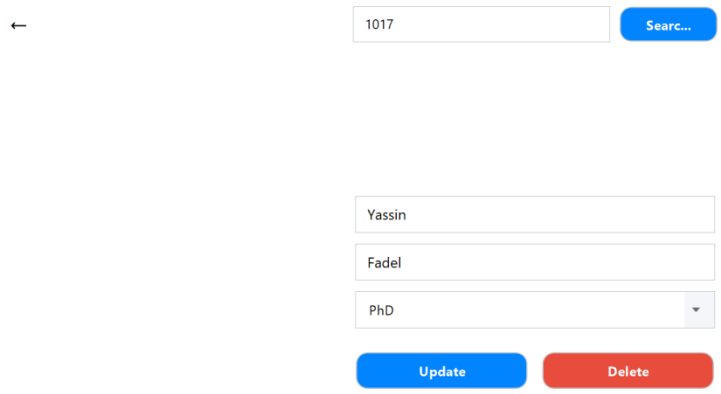


Figure4. 9 : Cloud Storage Query Panel (CloudSearchForm)

This screen is used to search for and manage records within the cloud database environment. It requires the user to enter a unique Student ID, and upon execution, the interface retrieves and displays the student's details on the screen. From this interface, users can also modify or delete the retrieved data directly from the cloud database.

2.1.6. Blockchain Data Viewer

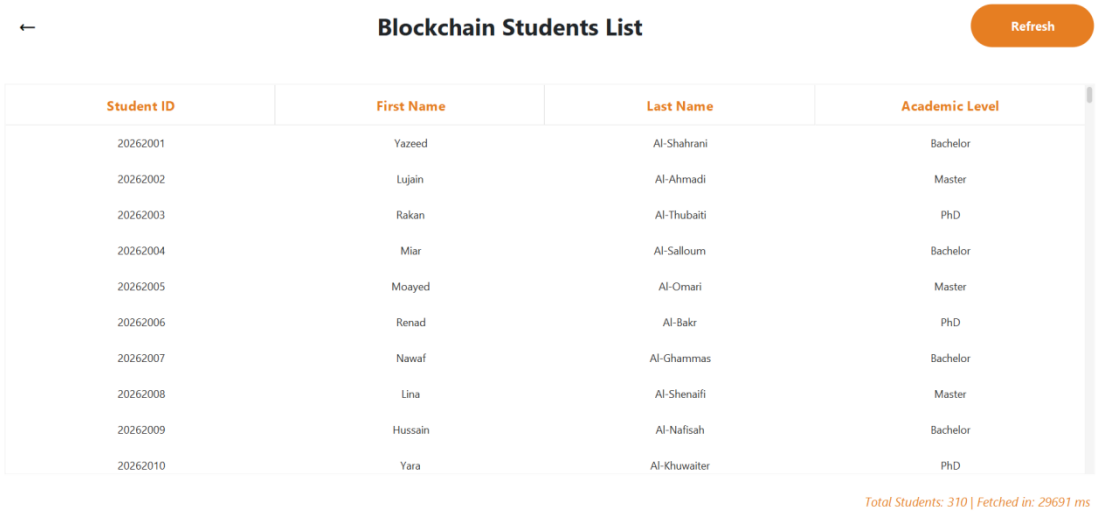


Figure4. 10 *: Decentralized Ledger Data Viewer*

This interface connects directly to the blockchain network to read and audit recorded student blocks. It fetches and parses the immutable, digitally signed transactions and displays them in a simple table format for clear tracking and monitoring. Similar to the cloud viewer, this screen calculates and measures the network response and display speed in milliseconds (ms) for performance comparison.

2.1.7. Blockchain Search Panel

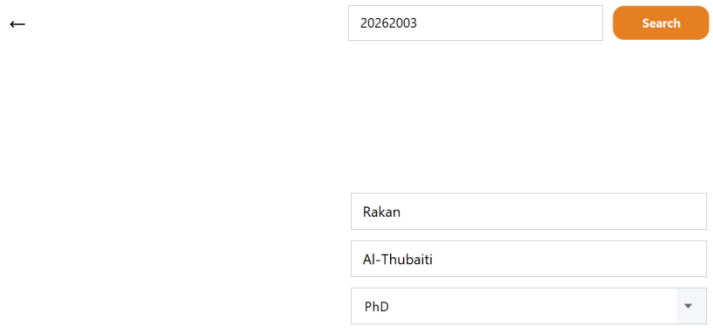


Figure4. 11 *: Ledger Search and Verification Interface*

This screen is dedicated to searching for and verifying records inside the blockchain using the Student ID, scanning transactions to confirm data integrity and digital validity.

2.2. Performance Benchmarking and Comparative Analysis

To evaluate the scalability, throughput, and system responsiveness of the proposed hybrid architecture, comprehensive empirical stress tests were performed. Execution latencies were tracked and recorded in milliseconds (ms) across two critical dimensions: Data Ingestion (Storage) and Data Retrieval (Display).

2.2.1. Data Ingestion Performance (Storage Tests)

The first test sequence evaluated the time taken to process and store varying volumes of student profiles simultaneously. The test workloads scaled up from a baseline of 10 records to a maximum of 150 records. Table 4.1 summarizes the collected latency metrics.

Data Volume (Student Records)	Cloud Ingestion Latency	Blockchain Ingestion Latency

10 Students	711 ms	1,885 ms
50 Students	2,386 ms	4,520 ms
100 Students	3,097 ms	8,601 ms
150 Students	4,545 ms	12,407 ms

Table4.1: Data Ingestion Performance (Storage Tests)

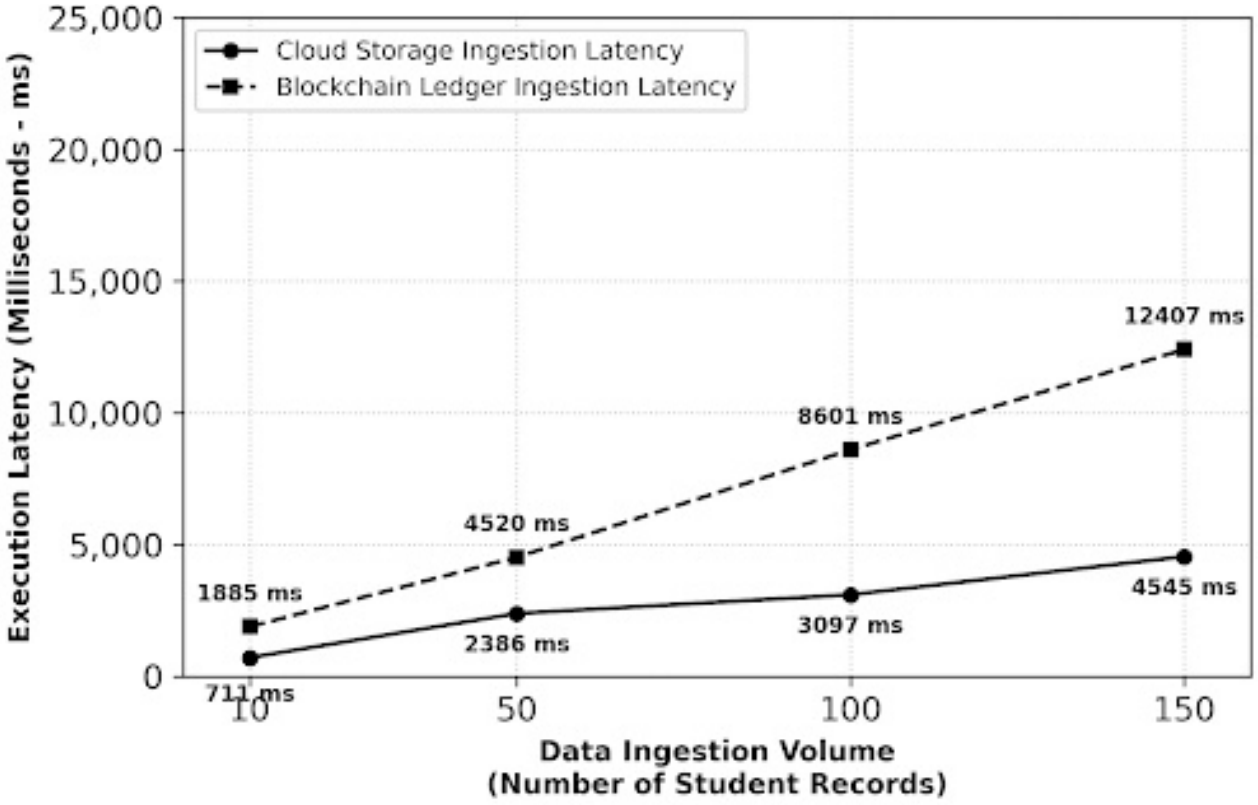


Figure4. 12: Data Ingestion Performance Curve (Storage)

2.2.2. Data Retrieval and Display Performance (Query Tests)

The second test sequence evaluated the system's ability to fetch, parse, and render records dynamically onto the visual user tables. This test captures environmental variances. Table 4.2 summarizes the results.

Display Volume (Student Records)	**Cloud Retrieval Latency**	**Blockchain Retrieval Latency**

10 Students	699 ms	715 ms
60 Students (Test Iteration A)	2,411 ms	3,900 ms
160 Students (Test Iteration B)	3433 ms	10,554 ms
310 Students (Mass View)	10,306 ms	20,181 ms

Table4. 2 Data Retrieval and Display Performance (Query Tests)

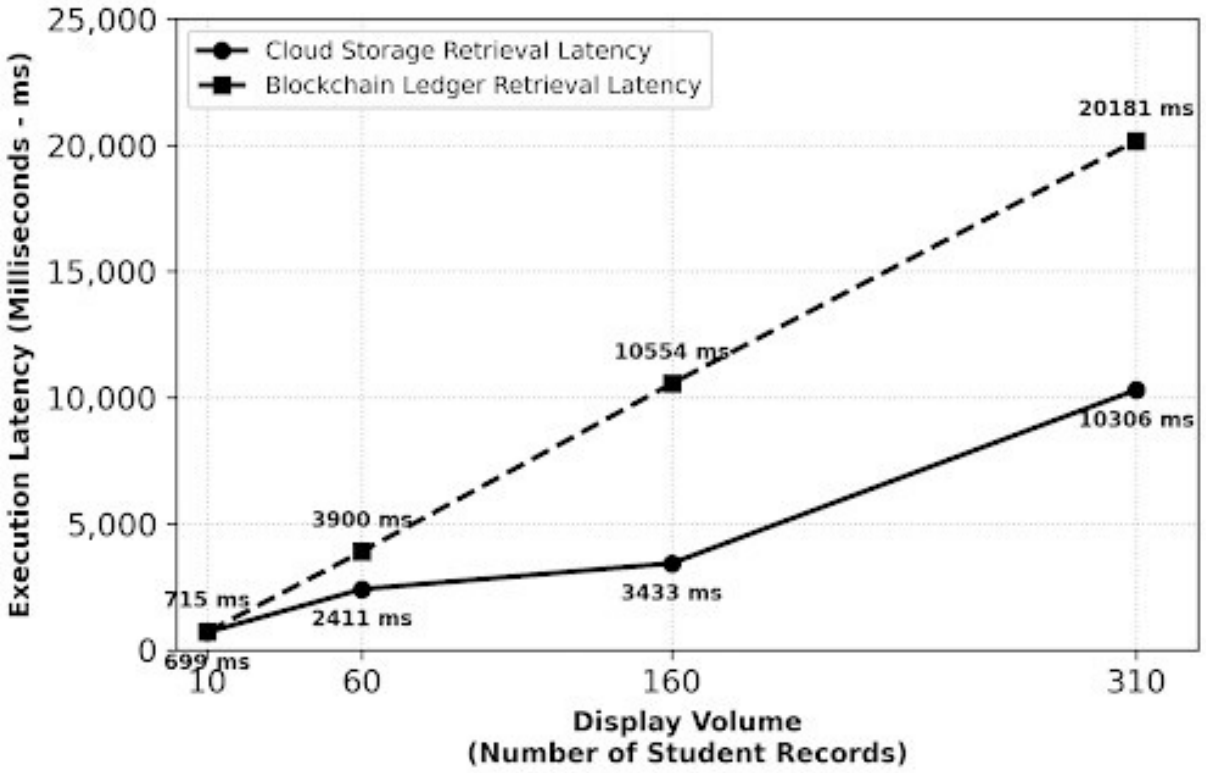


Figure4. 13 : Data Ingestion Performance curve (Storage)

2.2.3. Comprehensive Technical Interpretation

The empirical results collected from the performance stress tests reveal distinct

operational behaviors between the cloud storage layer and the blockchain ledger architecture during data ingestion and retrieval phases.

- **Scalability and Latency Trends in Data Ingestion:** The data measured in the storage tests demonstrates a stable linear growth in execution time for both environments as the volume scales from 10 to 150 student records. The cloud layer consistently maintains lower latency, moving from 711 ms to 4,545 ms. In comparison, the blockchain ledger exhibits a higher latency overhead, climbing from 1,885 ms and reaching 12,407 ms at 150 records. This performance gap is directly tied to the infrastructure design; the cloud processes straightforward write operations, while the blockchain infrastructure requires extra execution time to process cryptographic signatures and package transactions into blocks across the network nodes.

- **Efficiency and Indexing in Data Retrieval:** During the display query tests, the system evaluated data retrieval speeds across volumes reaching up to 310 student records. The cloud storage layer shows a significant speed advantage, retrieving 160 records in 3,433 ms and 310 records in 10,306 ms due to its optimized database indexing. The blockchain layer requires 10,554 ms for 160 records and reaches 20,181 ms at 310 records. This behavior occurs because retrieving and displaying data from a decentralized ledger requires scanning sequential blocks, parsing the transaction pipeline, and verifying cryptographic block integrity before rendering the data on screen.

- **Architectural Trade-offs and System Resilience:** The benchmarking results highlight a clear technical trade-off between centralized and decentralized infrastructures. While the cloud storage pipeline provides faster throughput and shorter response times for daily operations, the higher latency observed in the blockchain ledger is the necessary computational cost for ensuring absolute data immutability and tamper-proof academic records. Implementing both layers within the system achieves an optimal balance: the cloud layer delivers high

performance for frequent queries and acts as a rapid fallback, while the blockchain layer provides a secure, permanent audit trail for verification.

Conclusion

In conclusion, this chapter successfully demonstrated the practical implementation of the proposed system and evaluated its efficiency in both cloud and blockchain environments. The benchmarking results revealed a clear performance trade-off between the two technologies; while cloud storage exhibited high throughput and superior speed in data retrieval and processing, blockchain storage required significantly more execution time due to cryptographic operations and network consensus validation. However, this latency represents a necessary computational cost for the absolute security and data immutability that blockchain provides, proving that integrating both layers delivers an optimal system that balances performance and security.

Conclusion

General Conclusion

This dissertation focused on the study and comparison of Blockchain and Cloud Computing, two key technologies that have significantly influenced the development of modern information systems. The main objective was to analyze their security aspects and evaluate their ability to protect data and ensure trust in digital environments. To achieve this objective, both technologies were examined from theoretical and practical perspectives. A prototype application was implemented using MongoDB Atlas and BigchainDB to compare their behavior in terms of data storage, management, performance, and security. The obtained results showed that Cloud Computing provides high flexibility, scalability, and efficient resource management. However, its centralized nature may expose it to security and privacy concerns. In contrast, Blockchain offers stronger security guarantees through decentralization, immutability, transparency, and cryptographic protection, making data tampering and unauthorized modifications considerably more difficult. Based on the conducted comparison, it can be concluded that Blockchain provides a higher level of security, particularly regarding data integrity, traceability, and trust. Nevertheless, Cloud Computing remains an essential technology due to its performance, accessibility, and scalability advantages. Therefore, the objectives of this research have been successfully achieved. The study demonstrated that Blockchain can effectively enhance security and address several challenges associated with cloud environments. Furthermore, the integration of Blockchain and Cloud Computing represents a promising solution for building secure, reliable, and transparent information systems capable of meeting future technological demands. As future work, additional research

may focus on integrating smart contracts, improving access-control mechanisms, and evaluating the proposed solution in larger and more complex real-world environments.

List of References

List of References

[1]Labadi, H. (2024). The Use of Blockchain Technology in the Development of the Islamic Financial Market: Case Study: Smart Sukuk (Doctoral dissertation).

[2] Hamze, M. (2015). Autonomie, sécurité et QoS de bout en bout dans un environnement de Cloud Computing (Doctoral dissertation, Université de Bourgogne).

[3] MURTHY, Ch VNU Bharathi, SHRI, M. Lawanya, KADRY, Seifedine, et al. Blockchain based cloud computing: Architecture and research challenges. IEEE access, 2020, vol. 8, p. 20519020520-.

[4] Rahman, F. (2023). Optimizing blockchain performance with Innovative scalability solutions (Doctoral dissertation, Université Paris Cité).

[5] Netter, E. (2017). Numérique et grandes notions du droit privé (Doctoral dissertation, Université de Picardie-Jules Verne).

[6] BILAMI, Karam Eddine. Une approche légère basée BlockChain pour sécuriser les communications M2M dans l'IoT. 2024. Thèse de doctorat. Université de Haute Alsace-Mulhouse.

[7] KHAN, Abdul Ghaffar, ZAHID, Amjad Hussain, HUSSAIN, Muzammil, et al. « Un parcours du Web et de la blockchain vers l'industrie 4.0 : une vue d'ensemble ». Dans : Conférence internationale sur l'informatique innovante (ICIC) 2019. IEEE, 2019. P. 1-7.

[8] CHEN, Weiru. Exploring Blockchain-Based Digital Transformation In Organizations. 2022. Thèse de doctorat. Old Dominion University

[9] https://Litecoin.org

[10] https://Ethereum.org

[11] SAHI, Louis. Confiance et fiabilité du cycle de vie des données décentralisées: une approche basée sur la blockchain pour améliorer la traçabilité. 2025. Thèse de doctorat. Université de Toulouse.

[12] Delattre, B. (2018). Contribution à l'histoire de l'UFR des Sciences Pharmaceutiques et Biologiques de Nantes: l'École de plein exercice pendant la Première Guerre Mondiale (Doctoral dissertation, Université de Nantes, Unité de Formation et de Recherche de Sciences Pharmaceutiques et Biologiques).

[13] https://www.geeksforgeeks.org

[14] ZHANG, Zixu. Amélioration de l'efficacité et de la sécurité de la blockchain fragmentée . 2025. Thèse de doctorat. Université de technologie de Sydney (Australie).

[15] LEDUC, Guilain. Performance et sécurité d'une Blockchain auto-adaptative et innovante. 2022. Thèse de doctorat. Université de Lorraine.

[16] ALHEFDI, Mohammad. Un cadre pour la conception de systèmes de blockchain dans un contexte de ville intelligente . Université d'Alabama à Birmingham, 2022..

[17] BESAN|ON, Léo. Interopérabilité des Systèmes Blockchains. 2021. Thèse de doctorat. Université de Lyon.

[18] Sarmah, S. S. (2018). Understanding blockchain technology. Computer science and engineering, 8(2), 23-29.

[19] Yaga, D., Mell, P., Roby, N., & Scarfone, K. (2018). Draft blockchain technology overview (nistir-8202). National Institute of Standard and Technology, 59.

[20] Wilczynski, A. (2020). Blockchain-based task scheduling in computational clouds. Doctoral dissertation, PhD thesis.

[21] Zhang, R., Xue, R., & Liu, L. (2019). Security and privacy on blockchain. ACM Computing Surveys (CSUR), 52(3), 1-34.

[22] Imine, Y. (2019). Cloud computing security (Doctoral dissertation, Université de Technologie de Compiègne).

[23]Cornejo-Ramirez, M. (2016). Security for the cloud (Doctoral dissertation, Université Paris sciences et lettres)

[24] https://share.google

[25] Nelson, K. U. U. C. (2021). Security requirements for the application of blockchain technology in the mining industry: supply chain area (Doctoral dissertation, Master dissertation, Riga Technical University).

[26] Hamze, M. (2015). Autonomie, sécurité et QoS de bout en bout dans un environnement de Cloud Computing (Doctoral dissertation, Université de Bourgogne)

[27] Harauzek, D. (2022). Cloud Computing: Challenges of cloud computing from business users perspective-vendor lock-in.

[28] Medhioub, H. (2015). Architectures et mécanismes de fédération dans les environnements cloud computing et cloud networking (Doctoral dissertation, Institut National des Télécommunications).

[29] Aldribi, A. (2018). Cloud intrusion detection based on change tracking and a new benchmark dataset (Doctoral dissertation).

[30] Sen, J. (2019). Security and Security and Privacy Privacy Privacy Issues in Cloud Computing Computing. Innovation Labs, Tata Consultancy Services Ltd., Kolkata, INDIA.

[31] Lounis, A. (2014). Security in cloud computing (Doctoral dissertation, Université de Technologie de Compiègne).

[32] https://scalingo.com

[33] Chekired, D. A. (2019). Towards Efficient and Secure Cloud Architectures and Models: Application on Smart Grid, IoT and SDN Vehicular Networks (Doctoral dissertation, Université de Technologie de Troyes).
[34] Ghribi, C. (2014). Energy efficient resource allocation in cloud computing environments (Doctoral dissertation, Institut National des Télécommunications).
[35] mant10vbnjklmlkjhgfdfghjkl
[36] Khebbeb, K. (2019). Formalisation et évaluation de stratégies d'élasticité multi-couches dans le Cloud (Doctoral dissertation, Université de Pau et des Pays de l'Adour; Université Abdelhamid Mehri-Constantine 2 (Constantine, Algérie)).
[37] Taleb, H. (2019). Une architecture pour l'intégration du Cloud Computing dans les réseaux de capteurs sans fil (RCSF) (Doctoral dissertation, Université de Haute Alsace-Mulhouse; Université Hadj Lakhdar (Batna, Algérie)).
[38] Popa, D. (2013). Security of Mobile Cloud Applications (Doctoral dissertation, Université de Cluj-Napoca (TUCN)).
[39] Awan, I. A., Shiraz, M., Hashmi, M. U., Shaheen, Q., Akhtar, R., & Ditta, A. (2020). Secure framework enhancing AES algorithm in cloud computing. Security and communication networks, 2020(1), 8863345.
[40] Carvallo, P. (2018). Security in the Cloud: an anomaly-based detection framework for the insider threats (Doctoral dissertation, Université Paris Saclay (COmUE)).
[41] Bhayal, S. (2011). A study of security in cloud computing. California State University, Long Beach.
[42] Gupta, S. Cloud Computing Security Threats.
[43] Ahmadi, S. (2023). Security and privacy challenges in cloud-based data warehousing: a comprehensive review. International Journal of Computer Science Trends and Technology (IJCST), ISSN 2347.27-17 ,(6)11 ,8578–
[44] Alalmaie, A. (2023). Zero Trust With Guaranteed Accuracy Architecture Implementation for Intrusion Detection Systems (ZTA-IDS). University of Technology Sydney (Australia).
[45] N'Goran, K. R. (2023). Stratégie de sécurité Zero Trust dans un environnement de cloud communautaire (Doctoral dissertation, Ecole nationale supérieure Mines-Télécom Atlantique; Institut National Polytechnique Félix Houphouët-Boigny (Yamoussoukro, Côte d'Ivoire)).
[46] Zou, J., He, D., Zeadally, S., Kumar, N., Wang, H., & Choo, K. R. (2021). Integrated blockchain and cloud computing systems: A systematic survey, solutions, and challenges. ACM Computing Surveys (CSUR), 54(8), 1-36.
[47] Zou, J., He, D., Zeadally, S., Kumar, N., Wang, H., & Choo, K. R. (2021). Integrated blockchain and cloud computing systems: A systematic survey, solutions, and challenges. ACM Computing Surveys (CSUR), 54(8), 1-36.
[48] Murthy, C. V. B., Shri, M. L., Kadry, S., & Lim, S. (2020). Blockchain based cloud computing: Architecture and research challenges. IEEE access, 8, 205190-205205.

[49]Eriksson, H. E., Penker, M., Lyons, B., & Fado, D. (2003). UML 2 toolkit. John Wiley & Sons.

[50] https://www.uml-diagrams.org

[51]Crétin, L. Travaux d'Etudes et de Recherche Développement d'un mini Analyseur Statique de code intégré dans Eclipse.

[52]Bouchenak, S. (2001). Mobilité et persistance des applications dans l'environnement JAVA (Doctoral dissertation, Institut National Polytechnique de Grenoble-INPG).

[53]Miftakhov, E., Mustafina, S., Kashnikova, A., & Akimov, A. (2024). Development of a cloud service for comprehensive research of polymer synthesis processes. Emerging Science Journal, 8(6), 2539-2553.

[54]Orjuela, K. G., Gaona-Garcیa, P. A., & Marin, C. E. M. (2021). RETRACTED ARTICLE: Towards an agriculture solution for product supply chain using blockchain: case study Agro-chain with BigchainDB. Acta Agriculturae Scandinavica, Section B-Soil & Plant Science, 71(1), 1-16.