

A Project report on

**FILE DATA SECURITY USING ELLIPTIC CURVE
CRYPTOGRAPHY**

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

BACHELOR OF TECHNOLOGY

in

**COMPUTER SCIENCE AND ENGINEERING
(DATA SCIENCE)**

By

B. SWAROOPA	214G1A32B1
B. ROHINI	214G1A3287
B. SUVARCHALA	214G1A32A9
K. YUGANDHAR	224G5A3215

Under the Guidance of

Mr. K. Venkatesh, M.Tech., (Ph.D)



**Department of Computer Science & Engineering
(Data Science)**

**SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY
(AUTONOMOUS)**

Rotarypuram Village, B K Samudram Mandal, Ananthapuramu - 515701

2024-2025

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

(AUTONOMOUS)

(Affiliated to JNTUA, Accredited by NAAC with 'A' Grade, Approved by AICTE, New Delhi &

Accredited by NBA (EEE, ECE & CSE)

Rotarypuram Village, BK Samudram Mandal, Ananthapuramu-515701

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING (DATA SCIENCE)



Certificate

This is to certify that the project report entitled **File Data Security using Elliptic Curve Cryptography** is the bonafide work carried out by **B. Swaroopa, B. Rohini, B. Suvarchala, K. Yugandhar** bearing Roll Number **214G1A32B1, 214G1A3287, 214G1A32A9, 224G5A3215** in partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science & Engineering (Data Science)** during the academic year 2024-2025.

Project Guide

Mr. K. Venkatesh, M.Tech., (Ph.D)

Assistant Professor

Head of the Department

Dr. P. Chitralingappa, M.Tech, Ph.D

Associate Professor

Date:

Place: Rotarypuram

External Examiner

DECLARATION CERTIFICATE

We students of Department Of Computer Science & Engineering (Data Science), **SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY (AUTONOMOUS)**, Rotarypuram, hereby declare that the dissertation entitled “**FILE DATA SECURITY USING ELLIPTIC CURVE CRYPTOGRAPHY**” embodies the report of our project work carried out by us during IV year under the guidance of Mr. K. Venkatesh, M.Tech.,Ph.D., Assistant Professor, Department of CSE (Data Science), Srinivasa Ramanujan Institute of Technology, and this work has been submitted for the partial fulfillment of the requirements for the award of degree of Bachelor of Technology.

The results embodied in this project report have not been submitted to any other University or Institute for the award of any Degree or Diploma.

Date:

Place:

S.No.	Name of the Student	Roll Number	Signature
1	B. Swaroopa	214G1A32B1	
2	B. Rohini	214G1A3287	
3	B. Suvarchala	214G1A32A9	
4	K. Yugandhar	224G5A3215	

Vision & Mission of the SRIT

Vision:

To become a premier Educational Institution in India offering the best teaching and learning environment for our students that will enable them to become complete individuals with professional competency, human touch, ethical values, service motto, and a strong sense of responsibility towards environment and society at large.

Mission:

- Continually enhance the quality of physical infrastructure and human resources to evolve in to a center of excellence in engineering education.
- Provide comprehensive learning experiences that are conducive for the students to acquire professional competences, ethical values, life-long learning abilities and understanding of the technology, environment and society.
- Strengthen industry institute interactions to enable the students work on realistic problems and acquire the ability to face the ever changing requirements of the industry.
- Continually enhance the quality of the relationship between students and faculty which is a key to the development of an exciting and rewarding learning environment in the college.

Vision & Mission of the Department of CSE (Data Science)

Vision:

To evolve as a leading department by offering best comprehensive teaching and learning practices for students to be self-competent technocrats with professional ethics and social responsibilities.

Mission:

- DM 1: Continuous enhancement of the teaching-learning practices to gain profound knowledge in theoretical & practical aspects of computer science applications.
- DM 2: Administer training on emerging technologies and motivate the students to inculcate self-learning abilities, ethical values and social consciousness to become competent professionals.
- DM 3: Perpetual elevation of Industry-Institute interactions to facilitate the students to work on real-time problems to serve the needs of the society.

Program Educational Objectives (PEOs)

An SRIT graduate in Computer Science & Engineering (Data Science), after three to four years of graduation will:

- PEO 1: Lead a successful professional career in IT / ITES industry / Government organizations with ethical values.
- PEO 2: Become competent and responsible computer science professional with good communication skills and leadership qualities to respond and contribute significantly for the benefit of society at large.
- PEO 3: Engage in life-long learning, acquiring new and relevant professional competencies / higher academic qualifications.

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of people who made it possible, whose constant guidance and encouragement crowned our efforts with success. It is a pleasant aspect that we have now the opportunity to express my gratitude for all of them.

It is with immense pleasure that we would like to express my indebted gratitude to my Guide **Mr. K. Venkatesh, M.Tech.,(Ph.D), Assistant Professor, Computer Science and Engineering**, who has guided me a lot and encouraged me in every step of the project work. We thank him for the stimulating guidance, constant encouragement and constructive criticism which have made possible to bring out this project work.

We are very much thankful to **Dr. P. Chitralingappa, M.Tech.,Ph.D, Associate Professor & Head of the Department, Computer Science and Engineering (Data Science)**, for his kind support and for providing necessary facilities to carry out the work.

We wish to convey my special thanks to **Dr. G. Bala Krishna, M.Tech.,Ph.D, Principal of Srinivasa Ramanujan Institute of Technology** for giving the required information in doing my project work. Not to forget, we thank all other faculty and non-teaching staff, and my friends who had directly or indirectly helped and supported me in completing my project in time.

We also express our sincere thanks to the Management for providing excellent facilities.

Finally, we wish to convey our gratitude to our family who fostered all the requirements and facilities that we need.

Project Associates

214G1A32B1

214G1A3287

214G1A32A9

224G5A3215

ABSTRACT

Encryption makes it possible to send private information via unprotected channels without worrying about data loss or modification by unauthorized parties. For data security in various contexts, many encryption algorithms have developed throughout time. This paper focuses on symmetric encryption, often used for key exchange due to its large key size. In particular, the AES (Advanced Encryption Standard) algorithm is widely adopted for securing sensitive data due to its efficiency and robust encryption. Additionally, the ECC (Elliptic Curve Cryptography) algorithm is explored for its lightweight yet secure approach to key exchange, offering an optimal balance between security and computational efficiency.

In the context of data centers and cloud computing, security is paramount. This paper proposes a quantum computing algorithm for encrypting cloud data. This combination leverages quantum computing's rapid complex computations. The quantum computing algorithm offers fast, efficient, and secure data protection in cloud environments, reducing computational power requirements while enhancing overall efficiency. Moreover, it addresses the limitations of traditional encryption methods, including AES and ECC, by providing a scalable solution for large-scale data encryption. By integrating quantum computing, AES, and ECC, we can achieve unprecedented levels of security and performance. The potential impact of this approach could revolutionize data protection standards in the cloud industry.

Keywords: Symmetric encryption, quantum computing, computational power

Contents

List of Figures	ix-x
List of Abbreviations	xi
Chapter-1: Introduction	
1.1 Motivation	1
1.2 Problem Statement	1
1.3 Objective of the Project	2
Chapter -2: Literature Survey	3
Chapter- 3: Planning	
3.1 Existing System	6
3.2 Disadvantages	6
3.3 Proposed System	7
3.3.1 Advantages	7
3.3.2 System Architecture	8
Chapter– 4: Requirement Analysis	
4.1 Functional and Non-functional requirements	9
4.2 Hardware Requirements	10
4.3 Software Requirements	10
4.4 Installation of Python	11
4.5 Installation of Visual Studio	13
Chapter– 5: Design	
5.1 UML Diagram	16
5.2 Class Diagram	16
5.3 Use Case Diagram	17
5.4 Sequence Diagram	17
5.5 Collaboration Diagram	18

5.6	Deployment Diagram	19
5.7	Activity Diagram	19
5.8	Component Diagram	21
5.9	ER Diagram	21
5.10	Data Flow Diagram	22
Chapter – 6: Implementation		
6.1	Modules	24
6.2	ECC: Elliptical curve cryptography	25
6.3	AES (Advanced Encryption Standard)	26
6.4	Quantum Computing	27
6.5	Quantum-Based Encryption in ECC Algorithm	29
6.5.1	Quantum Key Generation	29
6.5.2	Text to Binary Conversion	29
6.5.3	Encryption	30
6.5.4	Decryption	30
6.5.5	Advantages of Quantum Key Generation in ECC	31
Chapter-7: Result		32
Conclusion		35
References		36
Project Publication		

LIST OF FIGURES

Fig No	Description	page No
3.1	Work Flow of Proposed System	8
4.1	Python download website	11
4.2	Python executable installer	12
4.3	Python path addition	12
4.4	Additional Features	13
4.5	Setup Successful	13
4.6	Download Visual Studio	14
4.7	Visual Studio Download	14
4.8	Installing Visual Studio Code	15
4.9	Installation Completed	15
5.1	Class Diagram	16
5.2	Use Case Diagram	17
5.3	Sequence Diagram	18
5.4	Collaboration Diagram	18
5.5	Deployment Diagram	19
5.6	Activity Diagram	20
5.7	Component Diagram	21
5.8	ER Diagram	22
5.9	Data Flow Diagram	23
6.1	Elliptical Curve	26
7.1	Home Page	32
7.2	Register Page	32
7.3	Upload Files Page	33
7.4	Download Files Page	33

7.5	AES Algorithm Graph	34
7.6	ECC-Quantum Algorithm Graph	34

LIST OF ABBREVIATIONS

ECC	Elliptic Curve Cryptography
RNGs	Random Number Generators
QKD	Quantum Key Distribution
AES	Advanced Encryption Standard
CBC	Cipher Block Chaining
ECB	Electronic Codebook
RSA	Rivest-Shamir-Adleman
DFD	Data Flow Diagram
UML	Unified Modelling Language
IDE	Integrated development environment

CHAPTER - 1

INTRODUCTION

1.1 Motivation:

The ever-growing reliance on cloud computing has significantly increased the need for robust data protection mechanisms. With sensitive data being transferred and stored across diverse cloud infrastructures, the risk of unauthorized access, loss, and modification has escalated. Traditional encryption algorithms, such as AES and ECC, have been widely used to secure data in transit and at rest. However, as the volume of data increases, the limitations of these encryption methods, including computational overhead and scalability concerns, become more pronounced. Quantum computing has emerged as a promising solution to address these limitations. The motivation behind this work is to explore how quantum computing can enhance cloud data protection by providing faster, more efficient encryption algorithms capable of securing large-scale cloud environments. By integrating quantum computing with traditional encryption methods like AES and ECC, we can ensure that file data is not only secure but also processed efficiently, meeting the growing demands of modern data centers. The potential to revolutionize cloud data protection with quantum-enhanced encryption could set new standards for securing sensitive information in the digital age.

1.2 Problem Statement:

The increasing reliance on cloud computing for storing and transmitting sensitive data has made data security a critical concern. Traditional encryption techniques such as AES and ECC are widely used to ensure confidentiality and integrity; however, they have limitations in terms of computational efficiency, scalability, and the ability to handle vast volumes of data in modern cloud environments. As data volumes continue to grow, the performance of these algorithms is challenged, often requiring high computational power and time-consuming processes, especially in large-scale cloud infrastructures. Additionally, these methods are vulnerable to emerging technologies such as quantum computing, which could potentially break current encryption protocols. The problem is exacerbated by the

constant need for more secure, faster, and scalable encryption methods to meet the growing demands of cloud environments. Thus, there is an urgent need for a solution that enhances the performance and scalability of cloud encryption, addressing both the security challenges posed by quantum computing and the inefficiencies inherent in traditional cryptographic methods.

1.3 Objective of the Project:

1. The primary objective of this project is to enhance the security and efficiency of file data protection by integrating quantum computing with traditional cryptographic algorithms, namely AES and ECC.
2. The aim is to develop an encryption algorithm that leverages the power of quantum computing to provide faster encryption while maintaining the robustness and security of AES and ECC.
3. Additionally, the project aims to create a scalable encryption solution that can effectively manage vast volumes of cloud data without compromising security, setting new standards for data protection in the cloud industry.
4. The objective is to deliver an encryption method that offers both secure and efficient data protection, revolutionizing cloud security practices.

CHAPTER – 2

LITERATURE SURVEY

This Chapter gives a brief description of various journals related to various journals related to the File Data Security using Elliptic Curve Cryptography. The mentioned papers are studied in order to understand the advantaged and disadvantages and the various technologies they have used to implement the website.

[1] R. Lu, X. Yuan, and X. Lin “Homomorphic Encryption for Cloud Computing: An Overview” published in IEEE Communications Surveys & Tutorials, 2020

They explored the concept of homomorphic encryption and its applications in cloud computing. The paper addresses the challenges and potential of this encryption method to secure cloud data, offering an in-depth look at its advantages for protecting sensitive information while maintaining privacy. This paper offers a comprehensive overview of homomorphic encryption, a cryptographic technique that allows computations to be performed on encrypted data without requiring decryption. The paper discusses the various types of homomorphic encryption (such as fully homomorphic and partially homomorphic encryption) and their applications in cloud computing.

[2] Kyu-Seok Shim; Boseon Kim; Wonhyuk Lee “Research on Quantum Key, Distribution Key and Post-Quantum Cryptography Key Applied Protocols for Data Science and Web Security” published in IEEE Access in 2024

This paper on exploring the integration of Quantum Key Distribution (QKD) and Post-Quantum Cryptography (PQC) protocols to secure data and communication systems, addressing the vulnerabilities posed by quantum computing to classical encryption methods. But PQC uses larger key sizes.

Quantum Key Distribution (QKD): QKD leverages the principles of quantum mechanics to establish a shared secret key between two parties, ensuring that any eavesdropping attempts are detectable. This is achieved through the no-cloning

theorem, which states that quantum data cannot be copied without altering the original state.

Post-Quantum Cryptography (PQC): PQC refers to cryptographic algorithms designed to be secure against attacks by both classical and quantum computers. Unlike QKD, which relies on quantum mechanics, PQC is based on complex mathematical problems that are believed to be resistant to quantum attacks.

[3] V. S. Pendyala, S. M. Arafath, and S. R. Kulkarni "Elliptic Curve Cryptography for Real-Time Data Encryption in IoT and Cloud Computing" IEEE Internet Of things Access, 2023

The paper explores the application of **Elliptic Curve Cryptography (ECC)** as a highly efficient method for securing real-time data in **IoT** and **cloud computing environments**. It highlights ECC's ability to provide strong encryption with smaller key sizes, making it particularly suitable for IoT devices with limited resources. In the context of the Internet of Things (IoT) and cloud computing, ECC is particularly advantageous due to its efficiency and low resource requirements. IoT devices often have limited processing power and memory, making ECC's lightweight nature suitable for real-time data encryption. In cloud computing, ECC ensures secure data transmission and storage without imposing significant computational burdens.

[4] X. Kong, J. Wang, and Q. Ni "Efficient Data Security and Privacy-Preserving Scheme in Cloud Computing" in IEEE Access, 2022

Their work focuses on developing a privacy-preserving scheme for cloud data security that effectively balances efficiency with robust protection, ensuring that cloud user's sensitive information is secure without compromising system performance. Homomorphic Encryption: This advanced encryption method enables computations to be performed on ciphertexts, producing encrypted results that, when decrypted, match the outcome of operations performed on the original plaintexts. With the advent of quantum computing, integrating quantum-resistant security measures is becoming increasingly important. A novel privacy-preserving distributed multiparty data

outsourcing scheme for cloud computing with Quantum Key Distribution (QKD) has been proposed to enhance security against quantum threats.

[5] IBM, "Quantum-safe cryptography: How it affects your information in the cloud," IBM Think Blog, 2024.

IBM's article discusses the emerging cybersecurity challenges posed by quantum computing, particularly concerning data in the cloud. The piece emphasizes that current encryption methods used to protect data in motion and at rest could be compromised by large quantum computers with millions of fault-tolerant qubits. IBM advocates for the adoption of quantum-safe cryptographic algorithms to mitigate these risks, highlighting the importance of proactive measures to secure data against future quantum attacks.

[6] A. Chhabra and S. Arora "An Elliptic Curve Cryptography Based Encryption Scheme for Securing the Cloud Against Eavesdropping Attacks" by IEEE 3rd International Conference on Collaboration and Internet Computing, 2024

The proposed scheme uses ECC for key exchange and data encryption, offering high security with smaller key sizes, ensuring both confidentiality and integrity of data during storage and transmission. The paper emphasizes how ECC improves efficiency over traditional encryption methods like RSA while defending against modern security threats in the cloud. The use of ECC not only strengthens security but also minimizes computational resources required for encryption and decryption processes. This efficiency is crucial for cloud environments, where optimizing performance while maintaining robust security is a priority. By integrating ECC into the encryption scheme, the proposed method aims to enhance the security of data stored in the cloud. This approach is particularly effective against eavesdropping attacks, ensuring that unauthorized parties cannot intercept or decipher the data during transmission or storage.

CHAPTER-3

PLANNING

3.1 Existing System

The existing system for data security in cloud computing primarily relies on traditional encryption technique like AES. While these methods provide robust security, they often require large key sizes, leading to significant computational overhead and increased energy consumption. This is particularly problematic in large-scale cloud environments, where efficiency and scalability are critical. Additionally, the existing systems face challenges in balancing security with performance, especially when handling real-time data encryption for cloud services and IoT devices. One of the commonly used encryption algorithms in this context is Elliptic Curve Cryptography (ECC). ECC offers a more efficient alternative to RSA and AES, providing strong encryption with smaller key sizes, which significantly reduces computational requirements and enhances system performance. Despite its advantages, ECC also faces certain limitations in terms of compatibility with legacy systems and its computational efficiency when applied to larger datasets in cloud environments. As a result, there is a growing need for more efficient encryption methods that can maintain high security without compromising system performance.

3.2 Disadvantages

- **Computational Overhead with AES:** While AES provides strong security, it often requires large key sizes, which can lead to high computational overhead and increased processing time, especially in large-scale cloud environments.
- **Energy Consumption:** The large key sizes used in AES result in higher energy consumption, making it less efficient for cloud systems, particularly when dealing with high-volume data and resource-constrained IoT devices.
- **Scalability Challenges:** The existing encryption algorithms like AES and ECC face difficulties when scaling to large datasets or high-throughput cloud environments, where maintaining encryption performance becomes increasingly difficult.

- **Compatibility Issues with Legacy Systems:** ECC, while more efficient than RSA, still faces compatibility issues with older systems and software that are not designed to support elliptic curve cryptography, limiting its adoption.
- **Vulnerability to Quantum Computing:** Both AES and ECC, while secure against classical attacks, are susceptible to quantum computing threats, which could potentially break their cryptographic security models and expose sensitive data in the future.

3.3 Proposed System

The proposed system introduces Elliptic Curve Cryptography (ECC) as an advanced encryption technique for enhancing data security in cloud computing environments. Unlike traditional methods, ECC offers strong encryption with significantly smaller key sizes, reducing computational overhead and energy consumption. This system aims to implement ECC for both data storage and transmission in cloud environments, ensuring robust security while maintaining high efficiency. In addition, the system will also integrate AES (Advanced Encryption Standard) to provide a hybrid encryption approach, leveraging the strength of both ECC and AES. AES, known for its security and wide acceptance, will be used for symmetric encryption in conjunction with ECC for asymmetric key exchange, further enhancing data protection. Furthermore, the proposed system incorporates Quantum Computing capabilities to address the future challenges posed by classical cryptographic methods. Quantum computing offers the potential to accelerate encryption processes and strengthen security, while also mitigating the risks that AES and ECC face against quantum-based attacks. By combining ECC, AES, and quantum computing, the system seeks to provide a more secure, scalable, and future-proof solution for modern cloud infrastructures, enhancing overall performance and ensuring data integrity across dynamic cloud services and IoT devices.

3.3.1 Advantages

- **Enhanced Security with Hybrid Encryption:** By combining Elliptic Curve Cryptography (ECC) for asymmetric key exchange and Advanced Encryption Standard (AES) for symmetric encryption, the proposed system offers robust

encryption. This hybrid approach leverages the strengths of both algorithms, ensuring higher security for cloud data storage and transmission.

- **Reduced Computational Overhead:** ECC utilizes smaller key sizes while maintaining a high level of security, significantly reducing computational overhead compared to traditional encryption methods like RSA and AES. This makes the system more efficient, particularly in large-scale cloud environments and resource-constrained IoT devices.
- **Scalability:** The proposed system is designed to scale seamlessly with cloud infrastructures. By optimizing encryption performance, it ensures that even large datasets and high-throughput environments can be efficiently encrypted, making it suitable for both current and future cloud services.
- **Real-Time Data Protection:** The system's ability to implement real-time encryption is crucial for dynamic cloud environments and IoT applications. This ensures that data is always protected during transmission and storage without affecting performance, thus providing continuous security in cloud and IoT ecosystems.

3.3.2 System Architecture

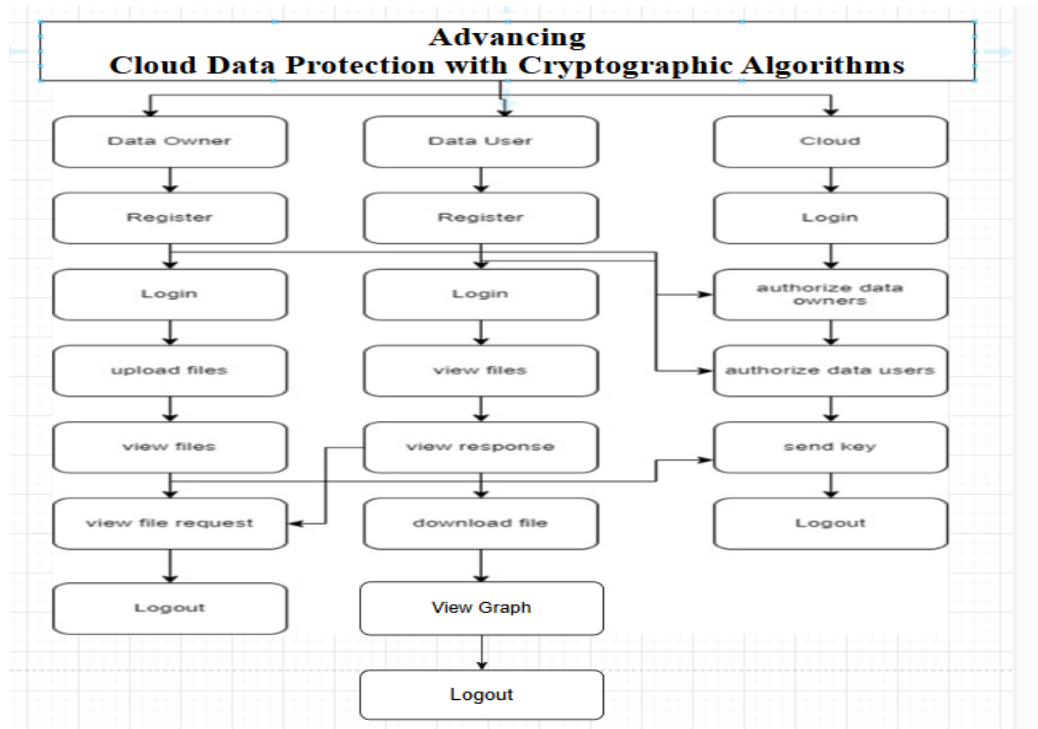


Fig. No. 3.1: Work Flow of Proposed System

CHAPTER – 4

REQUIREMENT ANALYSIS

4.1 Functional and Non-functional requirements

Requirement's analysis is very critical process that enables the success of a system or software project to be assessed. Requirements are generally split into two types: Functional and non-functional requirements.

Functional Requirements: The proposed system requires functionality for encrypting cloud data using a combination of AES, ECC, and Quantum Computing algorithms. These algorithms must provide secure data storage and transmission while ensuring fast encryption and decryption operations. The system should support real-time encryption for dynamic cloud environments and IoT devices. Additionally, the system must handle key management for both ECC and AES encryption methods. It should be capable of scaling to handle large datasets in cloud environments and support efficient encryption of sensitive data while maintaining high performance. These are the requirements that the end user specifically demands as basic facilities that the system should offer.

1. Data Owner – Register, Login, Upload files, View files, View file request, Logout.
2. Data User - Register, Login, View files, View response, Download file, Logout.
3. Cloud – Login, View and authorize Data owner, View and authorize Data user, Send key, Logout.

Non-functional requirements: The system must ensure high availability, scalability, and efficiency when encrypting large-scale data. It should minimize computational power and energy consumption while maintaining robust security levels. Additionally, the system needs to support easy integration with existing cloud infrastructure. The encryption process should have minimal latency to ensure real-time data encryption and quick responses. The system must also offer reliability and fault tolerance, ensuring that data remains protected even during hardware or software

failures. User experience should be seamless with minimal manual intervention required for managing encryption keys.

☐ Performance

- Measures how fast and efficiently the system responds under specific conditions.
- Includes response time, throughput, and resource utilization.
- Example: A web application should load a page within 2 seconds under normal traffic.

☐ Scalability

- Determines how well the system can handle increased workloads or users.
- Can be vertical (upgrading hardware) or horizontal (adding more servers).
- Example: An e-commerce platform should handle traffic spikes during sales events.

☐ Security

- Ensures protection against threats like unauthorized access, data breaches, or cyberattacks.
- Includes authentication, encryption, and access control mechanisms.
- Example: A banking app must use multi-factor authentication for user logins.

☐ Availability

- Defines the system's uptime and ability to remain operational without failures.
- Often measured using SLA (Service Level Agreement) metrics like 99.99% uptime.
- Example: A cloud-based SaaS application should ensure 24/7 availability with minimal downtime.

4.2 Hardware Requirements

Processor	- I3/Intel Processor
Hard Disk	- 160GB
Key Board	- Standard Windows Keyboard
Mouse	- Two or Three Button Mouse
Monitor	- SVGA
RAM	- 8GB

4.3 Software Requirements

Operating System	: Windows 7/8/10
Server side Script	: HTML, CSS, Bootstrap & JS
Programming Language	: Python
Libraries	: Django, Pandas, Mysql.connector, OS, Smtplib, Numpy
IDE/Workbench	: VS Code
Technology	: Python 3.6+
Server Deployment	: Xampp Server
Database	: MySQL

4.4 Installation of Python

Python has become one of the most popular programming languages of the 21st century. It is being used for multiple purposes in various sectors of business. Developers use Python for building applications and developing websites. Data Engineers use python for performing data analysis, statistical analysis, and building machine learning models. However, you can check if it exists on the system by running one line of command on the command prompt: `python--version`.

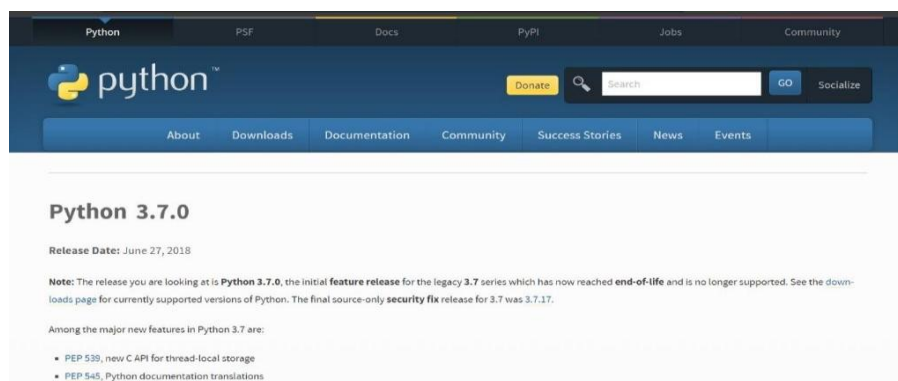
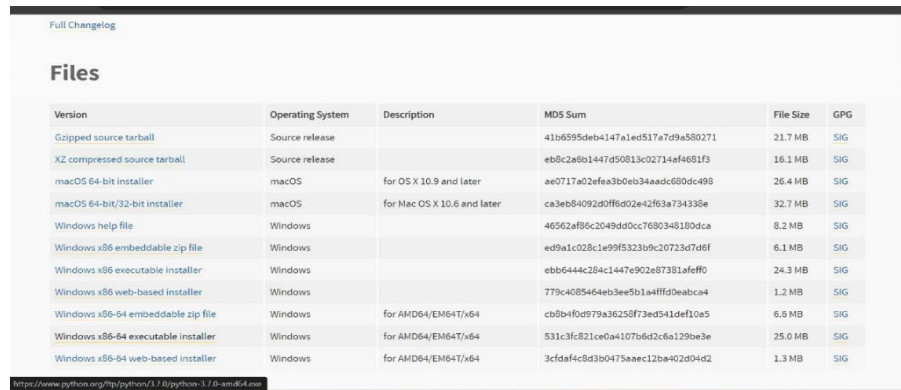


Fig. No. 4.1: Python download website

To download python, open the chrome and browse by typing python download. Download “python 3.7.0” version from the internet because it is compatible with our project.



Full Changelog

Files

Version	Operating System	Description	MD5 Sum	File Size	GPG
Gzipped source tarball	Source release		41b6595deb4147a1ed517a7d9a580271	21.7 MB	SIG
XZ compressed source tarball	Source release		eb8c2a8b1447d50813c02714af4681f3	16.1 MB	SIG
macOS 64-bit installer	macOS	for OS X 10.9 and later	ae0717a02fe3b0eb34aad680dc498	26.4 MB	SIG
macOS 64-bit/32-bit installer	macOS	for Mac OS X 10.6 and later	ca3eb54092d0ff6d02e42f63a734335e	32.7 MB	SIG
Windows help file	Windows		46562af86c2049dd0cc7680348180dca	8.2 MB	SIG
Windows x86 embeddable zip file	Windows		ed9a1c028c1e99f323b9c20723d7def	6.1 MB	SIG
Windows x86 executable installer	Windows		ebb6444c284c1447e902e87381afeff0	24.3 MB	SIG
Windows x86 web-based installer	Windows		779c408546eb3ee5b1a4fffd0eabca4	1.2 MB	SIG
Windows x86-64 embeddable zip file	Windows	for AMD64/EM64T/x64	cb8b4f0d979a36258f73ed541def10a5	6.6 MB	SIG
Windows x86-64 executable installer	Windows	for AMD64/EM64T/x64	531c3fc821ce0a4107b6d2c6a129be3e	25.0 MB	SIG
Windows x86-64 web-based installer	Windows	for AMD64/EM64T/x64	3cfdaf4c8d3b0475aaec12ba402d04d2	1.3 MB	SIG

<https://www.python.org/ftp/python/3.7.0/python-3.7.0-amd64.exe>

Fig. No. 4.2: Python executable

installer

Select the “windows x36-64 executable installer”.

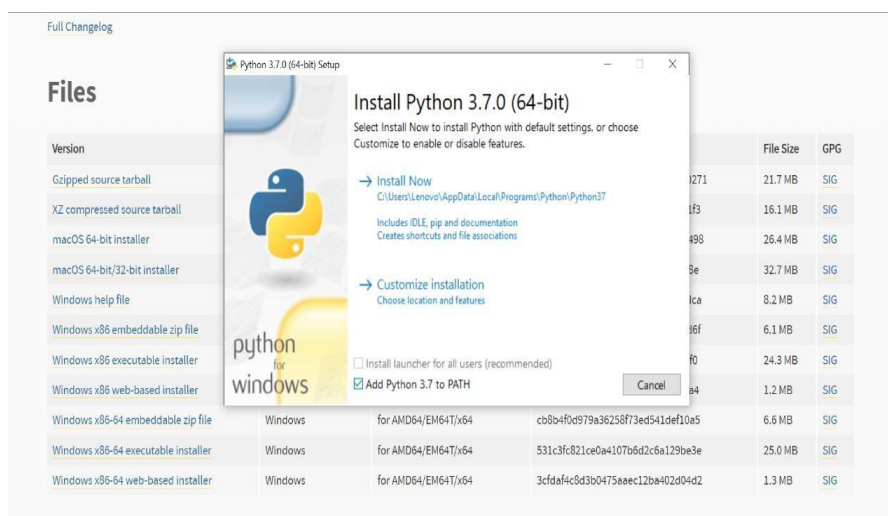


Fig. No. 4.3: Python path addition

Upon selecting the suitable version, a dialogue box is popped showing the options “install now” and “customize Installation”. Select the “Customize Installation” and check box the “Add python 3.7 to PATH”.

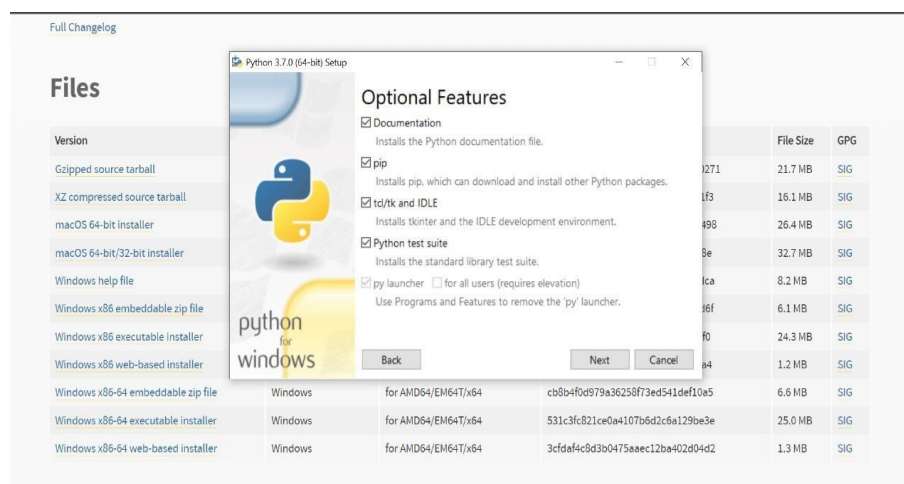


Fig. No. 4.4: Additional Features

Adding all the additional features that are required for the execution of the code. Click on “Next” button after selecting the features. Another dialogue box names Advanced Options will be displayed. Check marks the box “Install for all Users”. We can also change the location where the python software to be installed. It is preferred to choose the default location option for installation. Click on “Install” button for successful installation of Python. Python software is installing all the executable files and libraries.

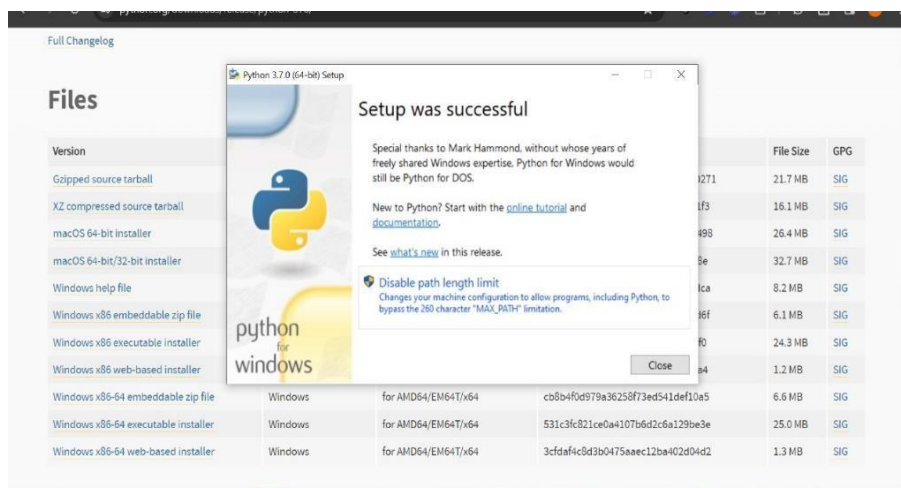


Fig. No. 4.5: Setup Successful

Python was successfully installed. Click on the “close” button to close the dialogue box.

4.5 Installation of Visual Studio

Visual Studio is an integrated development environment (IDE) developed by Microsoft. It provides comprehensive tools and features for software development across various platforms, including web, mobile, desktop, cloud, and more. Installing Visual Studio is essential for developers looking to leverage its powerful capabilities for writing, debugging, and deploying applications efficiently.

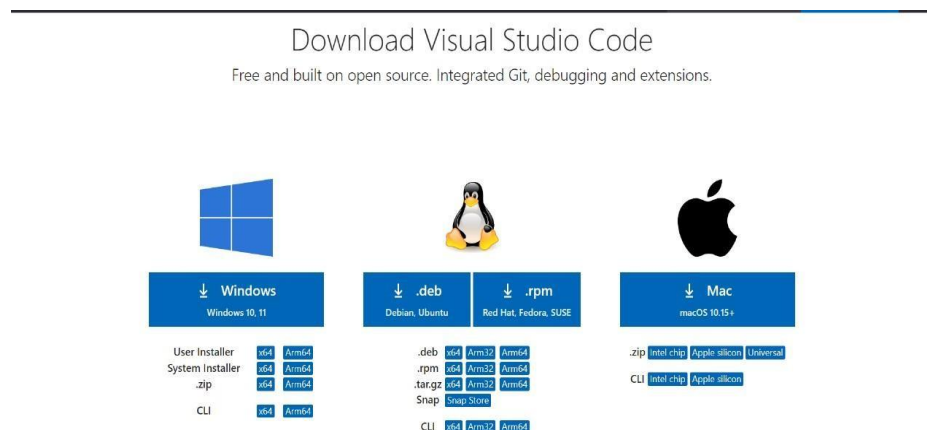


Fig. No. 4.6: Download Visual Studio

As Visual Studio is available for many Operating Systems, we have to select the type of the operating system that is supported by the Computer/Laptop.

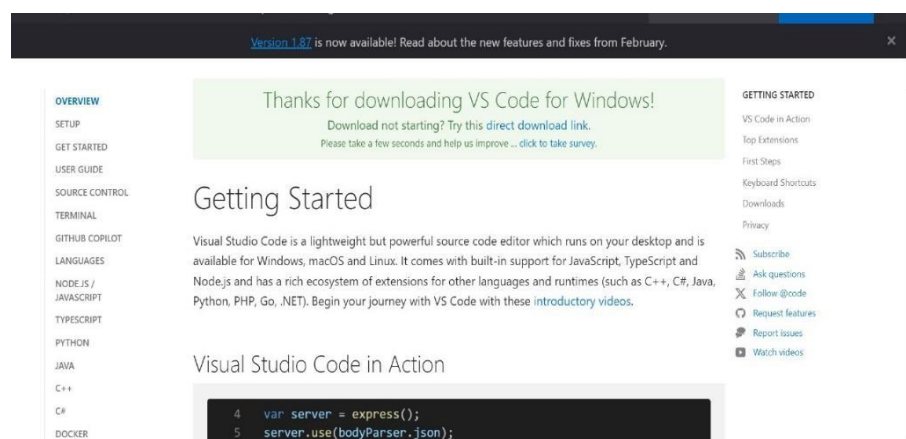


Fig. No. 4.7: Visual Studio Download

For setting up the Visual Studio Code, select the location to where it has to be installed. We can change the location by clicking on “Browse” button. Click on the “Next” button after selecting the location. Selecting all the additional features that are required for the successful installation of the visual studio code and click on “Next”. A Dialogue is shown displaying the Destination Location, Menu Folder and Additional Tools for the confirmation. Proceed by clicking on the “Install” button to install Visual Studio Code.

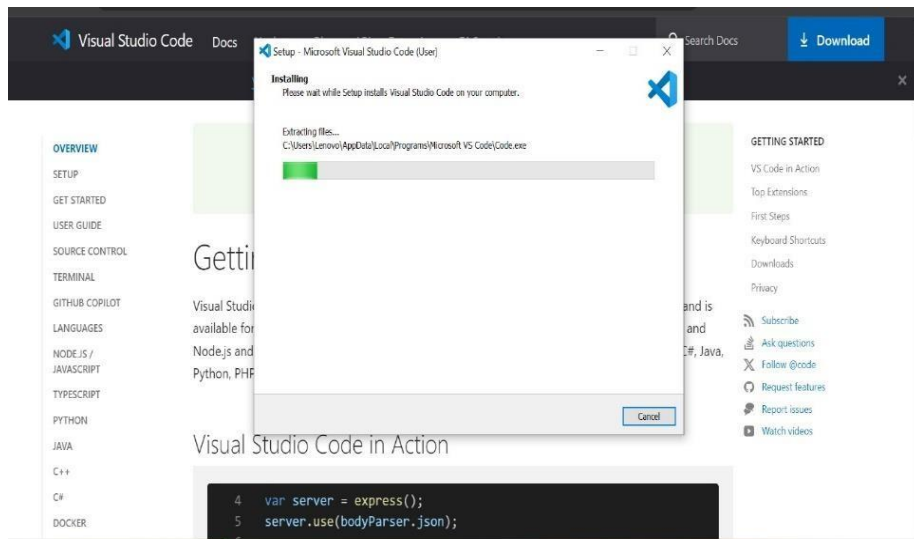


Fig. No. 4.8: Installing Visual Studio Code

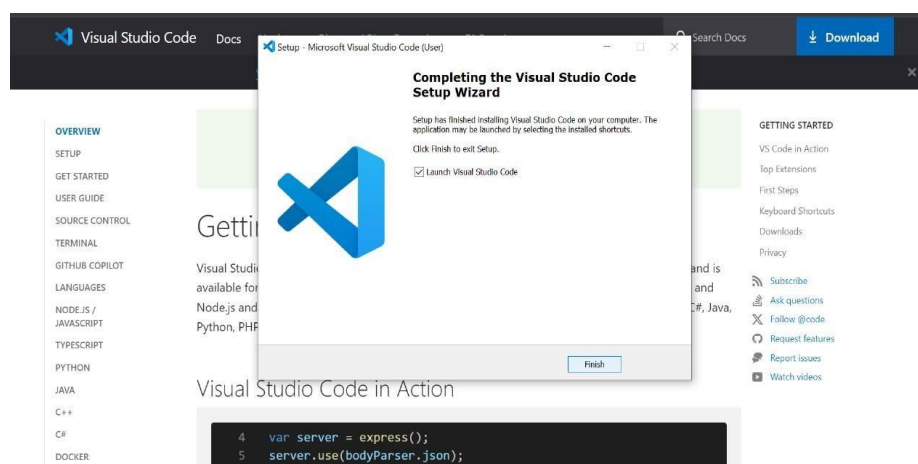


Fig. No. 4.9: Installation Completed

Visual Studio Code is installed.

CHAPTER – 5

DESIGN

5.1 UML Diagram:

UML, or Unified Modeling Language, is a standardized modeling language used in software engineering to visually represent systems, structures, behaviors, and processes. It provides a common and widely accepted way to visualize and communicate various aspects of software systems throughout their development lifecycle. UML diagrams serve as blueprints for understanding, designing, and documenting software systems.

5.2 Class Diagram:

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

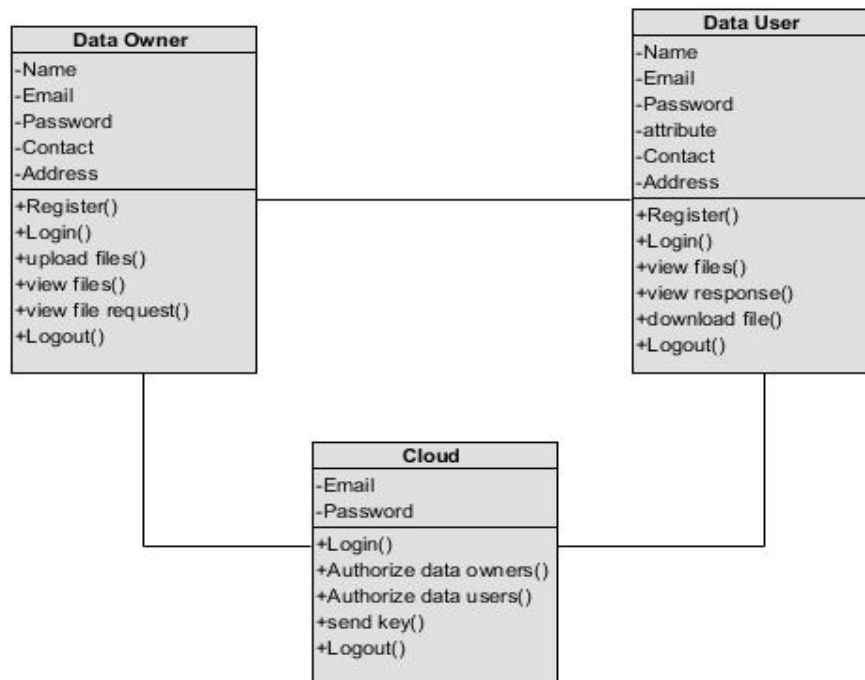


Fig. No. 5.1: Class Diagram

5.3 Use Case Diagram:

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

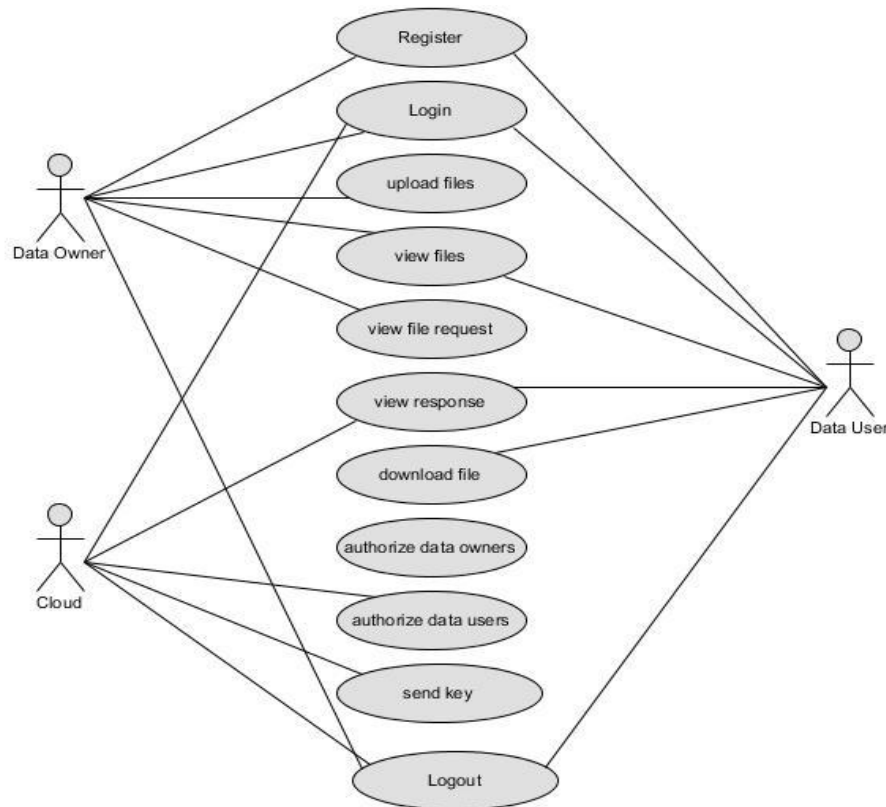


Fig. No. 5.2: Use Case Diagram

5.4 Sequence Diagram:

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

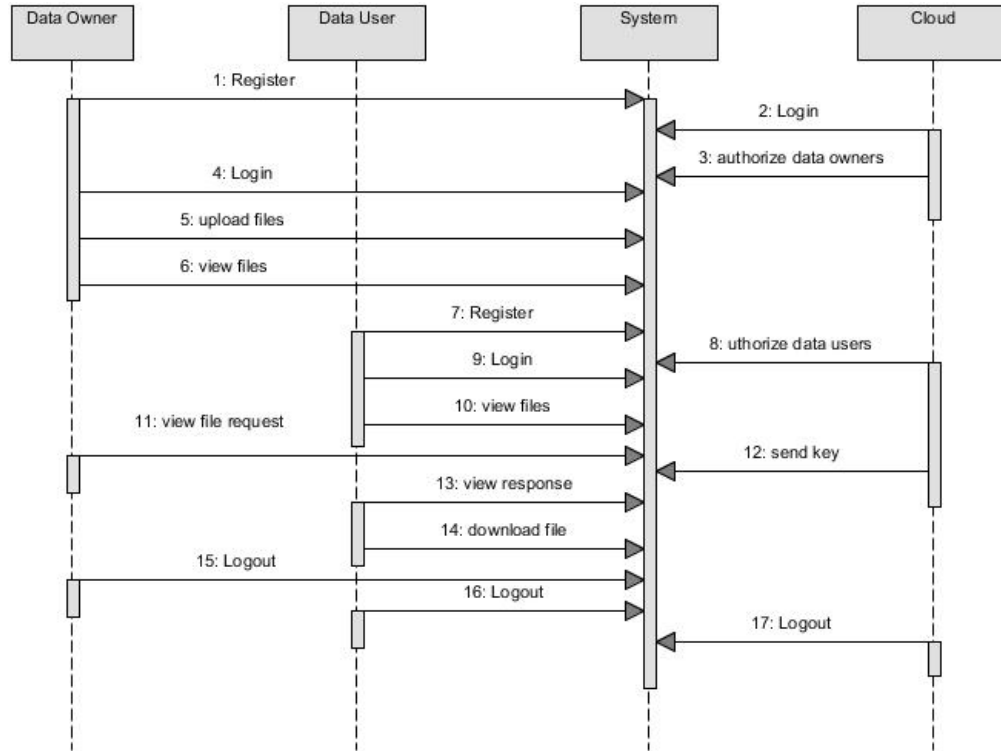


Fig. No. 5.3: Sequence Diagram

5.5 Collaboration Diagram:

In collaboration diagram the method call sequence is indicated by some numbering technique as shown below. The number indicates how the methods are called one after another. We have taken the same order management system to describe the collaboration diagram. The method calls are similar to that of a sequence diagram. But the difference is that the sequence diagram does not describe the object organization whereas the collaboration diagram shows the object organization.

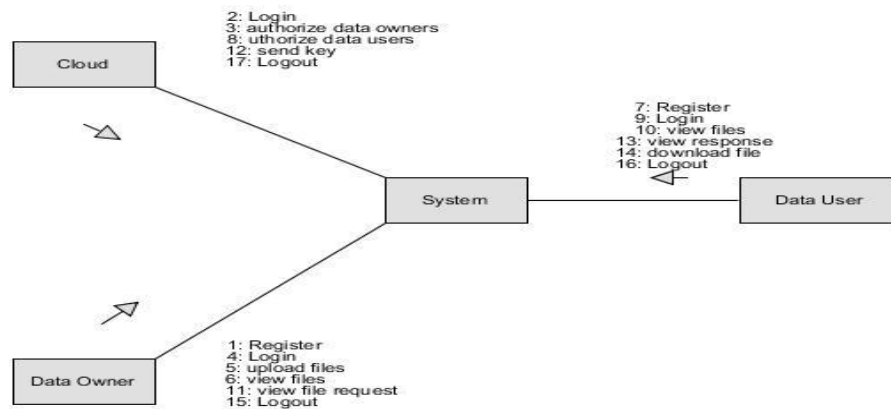


Fig. No. 5.4: Collaboration Diagram

5.6 Deployment Diagram:

Deployment diagram represents the deployment view of a system. It is related to the component diagram. Because the components are deployed using the deployment diagrams. A deployment diagram consists of nodes. Nodes are nothing but physical hardware's used to deploy the application.

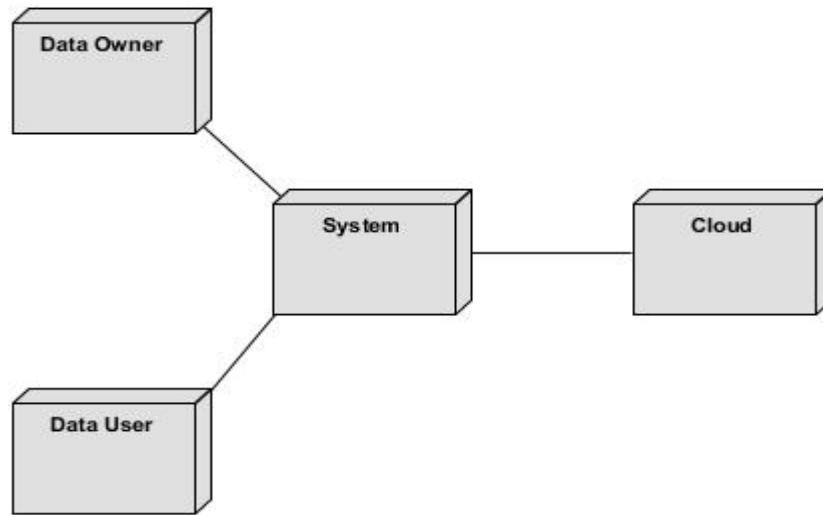


Fig. No. 5.5: Deployment Diagram

5.7 Activity Diagram:

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

Data Owner:

- Register: Allows the data owner to sign up on the platform by providing necessary details.
- Login: Enables the data owner to access the platform using their credentials after receiving authorization from the cloud administrator.
- Upload Files: Permits the data owner to upload files to the platform and encrypt them using quantum encryption algorithms.

- View File Requests: Enables the data owner to see requests from data users and decrypt files as needed.

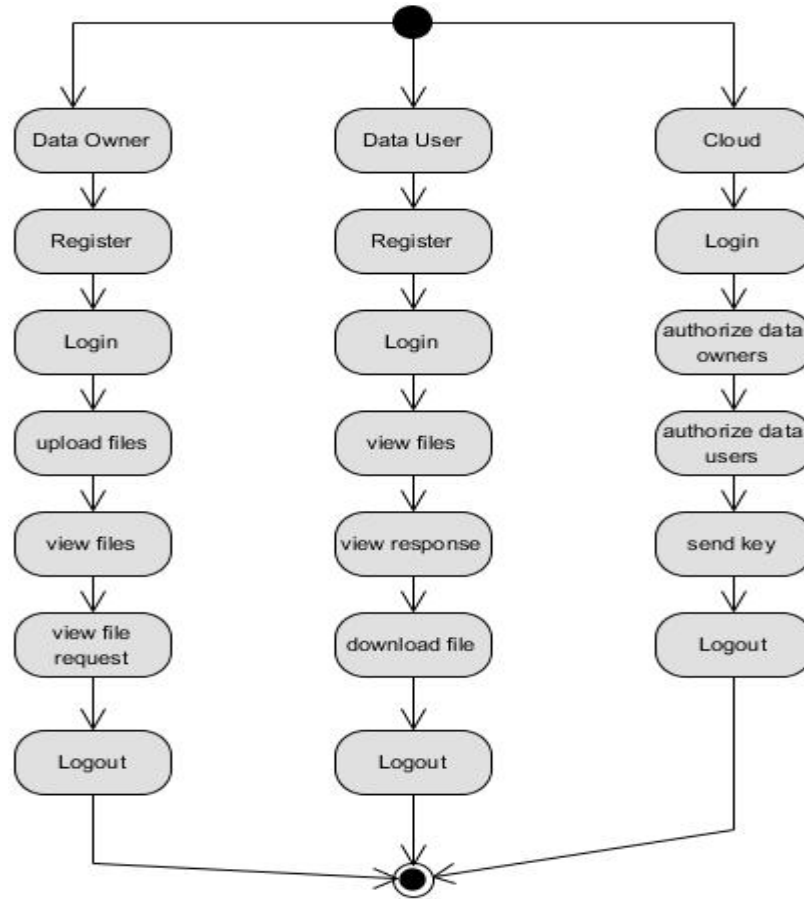


Fig. No. 5.6: Activity Diagram

Data User:

- Register: Allows the data user to create an account on the platform by entering required information.
- Login: Enables the data user to access the platform using their credentials after cloud administrator authorization.
- View Files: Permits the data user to browse available files and send access requests to data owners.
- View Responses: Allows the data user to check the status of their file access requests.
- Download Files: Enables the data user to download files using the provided decryption keys.

Cloud Administrator:

- Authorize Data Owners: Enables the administrator to review and authorize or deauthorize registered data owners.
- Authorize Data Users: Permits the administrator to review and authorize or deauthorize registered data users.
- Send Keys: Allows the administrator to securely send decryption keys to data users via email for file access.

5.8 Component Diagram:

A component diagram, also known as a UML component diagram, describes the organization and wiring of the physical components in a system. Component diagrams are often drawn to help model implementation details and double-check that every aspect of the system's required functions is covered by planned development.

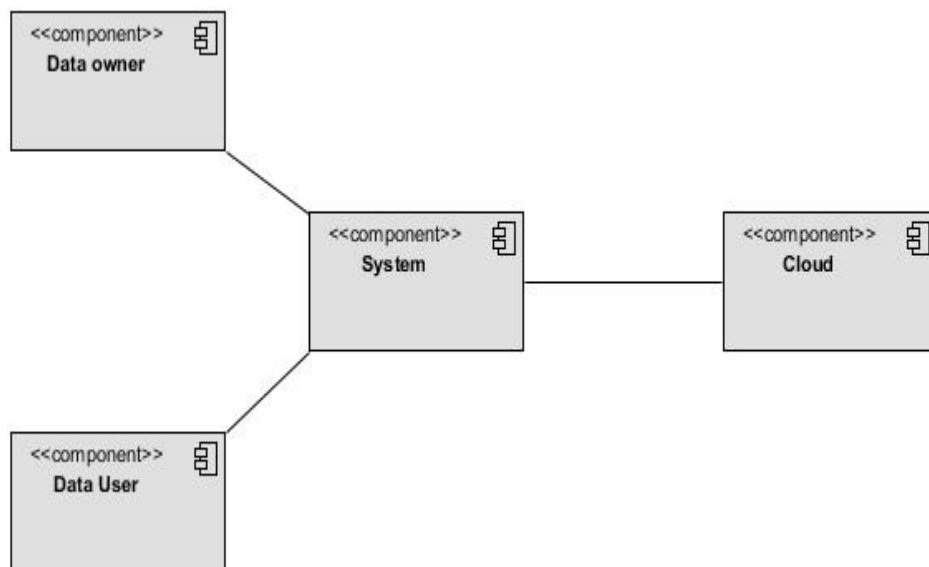


Fig. No. 5.7: Component Diagram

5.9 ER Diagram:

An Entity–relationship model (ER model) describes the structure of a database with the help of a diagram, which is known as Entity Relationship Diagram (ER Diagram). An ER model is a design or blueprint of a database that can later be implemented as a database. The main components of E-R model are: entity set and relationship set.

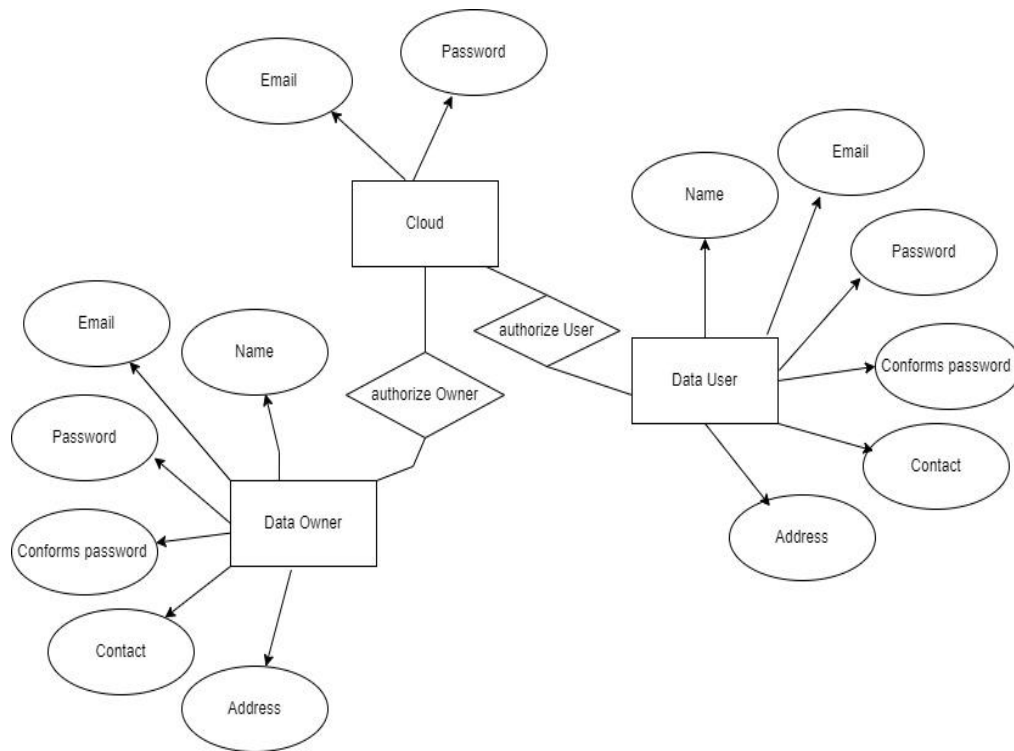


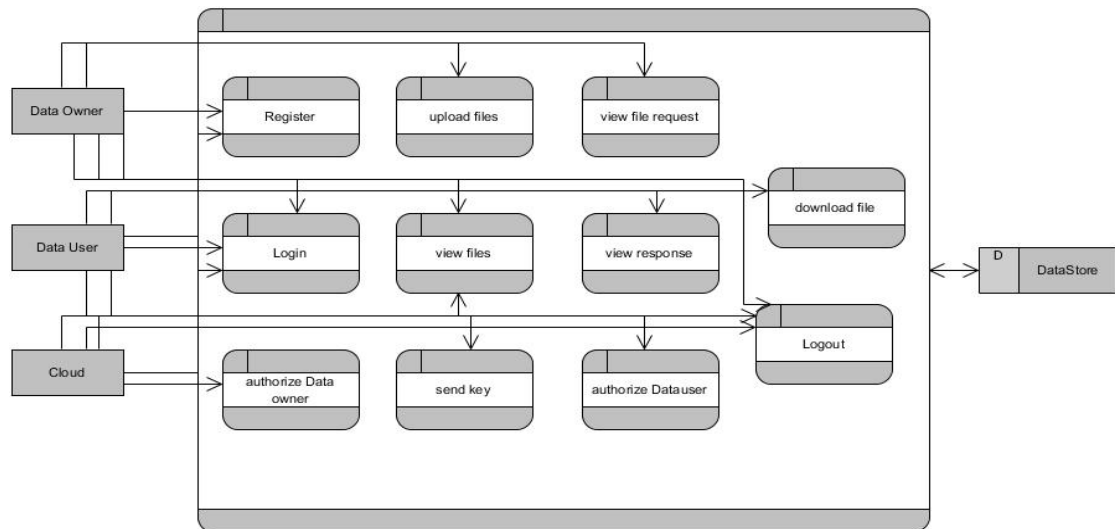
Fig. No. 5.8: ER Diagram

An ER diagram shows the relationship among entity sets. An entity set is a group of similar entities and these entities can have attributes. In terms of DBMS, an entity is a table or attribute of a table in database, so by showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database. Let's have a look at a simple ER diagram to understand this concept.

5.10 Data Flow Diagram:

A Data Flow Diagram (DFD) is a traditional way to visualize the information flows within a system. A neat and clear DFD can depict a good amount of the system requirements graphically.

It can be manual, automated, or a combination of both. It shows how information enters and leaves the system, what changes the information and where information is stored. The purpose of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communications tool between a systems analyst and any person who plays a part in the system that acts as the starting point for redesigning a system.

**Fig. No. 5.9: Data Flow Diagram**

CHAPTER - 6

IMPLEMENTATION

6.1 MODULES:

Data Owner:

- **Register:** Data owner can register into website by entering details.
- **Login:** He can login into website by entering their credentials.(after admin authorized)
- **Upload files:** Here he can upload their files into website and he can encrypt that file using ECC AES and Quantum Computing.
- **View files:** here he can view their files.
- **View file request:** Here Owner can view his file requests and decrypt the file.
- **Logout:** The Owner will logout from website.

Data User:

- **Register:** Data user can register into website by entering details.
- **Login:** He can login into website by entering their credentials.(after admin authorized)
- **View files:** He can view the file and send a file request to data owner.
- **View response:** here he can view the requested file response.
- **Download file:** he can download the file by using the key.
- **Logout:** The User will logout form here.

Cloud:

- **Login:** He can login into website by entering their credentials.
- **View and authorize Data owner:** here admin can view the registered owner and he can authorize and unauthorized the owner.
- **View and authorize Data user:** here admin can view the registered user and he can authorize and unauthorised the user.
- **Send key:** Here admin can send a key through email for downloading the file.
- **Logout:** The Cloud should be logout.

6.2 ECC: Elliptical curve cryptography

ECC is an alternative to the Rivest-Shamir-Adleman (RSA) cryptographic algorithm and is most often used for digital signatures in cryptocurrencies, such as Bitcoin and Ethereum, as well as one-way encryption of emails, data and software. An elliptic curve is not an ellipse, or oval shape, but it is represented as a looping line intersecting two axes, which are lines on a graph used to indicate the position of a point. The curve is completely symmetric, or mirrored, along the x-axis of the graph. Public key cryptography systems, like ECC, use a mathematical process to merge two distinct keys and then use the output to encrypt and decrypt data. One is a public key that is known to anyone, and the other is a private key that is only known by the sender and receiver of the data.

ECC generates keys through the properties of an elliptic curve equation instead of the traditional method of generation as the product of large prime numbers. From a cryptographic perspective, the points along the graph can be formulated using the following equation:

$$y^2 = x^3 + ax + b$$

ECC is like most other public key encryption methods, such as the RSA algorithm and Diffie-Hellman. Each of these cryptography mechanisms uses the concept of a one-way, or trapdoor, function. This means that a mathematical equation with a public and private key can be used to easily get from point A to point B. But, without knowing the private key and depending on the key size used, getting from B to A is difficult, if not impossible, to achieve.

ECC is based on the properties of a set of values for which operations can be performed on any two members of the group to produce a third member, which is derived from points where the line intersects the axes as shown with the green line and three blue dots in the below diagram labeled A, B and C. Multiplying a point on the curve by a number produces another point on the curve (C). Taking point C and bringing it to the mirrored point on the opposite side of the x-axis produces point D. From here, a line is drawn back to our original point A, creating an intersection at point E. This process can be completed n number of times within a defined max value. The n is the private key value, which indicates how many times the equation should be run, ending on the final

value that is used to encrypt and decrypt data. The maximum defined value of the equation relates to the key size used.

Elliptical curve showing three points of intersection

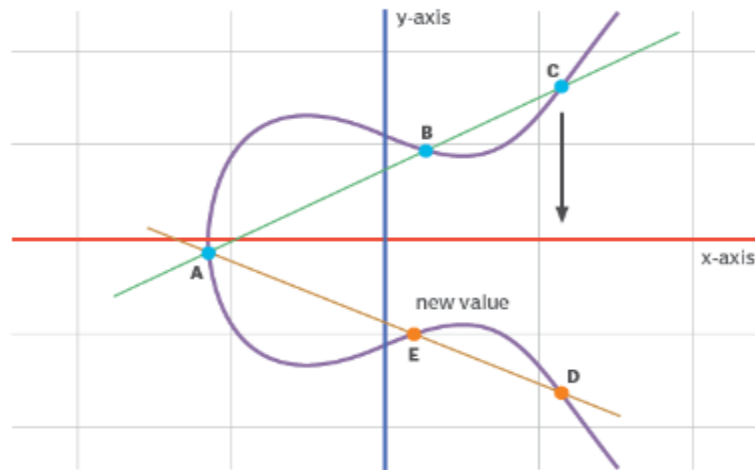


Fig. No. 6.1: Elliptical Curve

How secure is elliptical curve cryptography?

ECC is thought to be highly secure if the key size used is large enough. The U.S. government requires the use of ECC with a key size of either 256 or 384 bits for internal communications, depending on the sensitivity level of the information being transmitted.

But ECC is not necessarily any more or less secure compared to alternatives such as RSA. The primary benefit of ECC is the inherent efficiencies gained when encrypting and decrypting data.

6.3 AES (Advanced Encryption Standard):

The Advanced Encryption Standard (AES) is a symmetric-key block cipher that provides secure data encryption by operating on fixed-size blocks of data. AES supports three key sizes: 128, 192, and 256 bits, with the key size determining the number of rounds used in the encryption process—10 rounds for 128 bits, 12 rounds for 192 bits, and 14 rounds for 256 bits. AES begins with key expansion, where the input key is expanded into an array of round keys used during each encryption round. The

encryption process starts with an AddRoundKey step, where the input data (plaintext) is XORed with the first round key. In the subsequent rounds, several operations are performed on the data, including SubBytes, where each byte of the data block is substituted using a fixed substitution table (S-box), ShiftRows, which shifts the rows of the data block to introduce diffusion, and MixColumns, which mixes the data within each column to further spread out the data's influence across the entire block. Each round concludes with another AddRoundKey step, where the round key is XORed with the data. In the final round, the MixColumns step is omitted, and the remaining steps—SubBytes, ShiftRows, and AddRoundKey—are performed.

AES decryption is the reverse of encryption, with the inverse operations of SubBytes, ShiftRows, and MixColumns applied in reverse order to recover the original plaintext. The decryption also involves AddRoundKey in reverse order, using the round keys in the opposite sequence compared to encryption. AES is a block cipher, and to handle larger data sets, it operates with various modes, such as Electronic Codebook (ECB), where each block is independently encrypted, Cipher Block Chaining (CBC), which XORs each plaintext block with the previous ciphertext block, and Counter (CTR) mode, which uses a counter value combined with AES encryption to XOR with the plaintext. Padding is also used to ensure that the plaintext is a multiple of the block size (128 bits), with PKCS7 being a common padding scheme.

Key management is a critical aspect of AES security, requiring proper key generation, storage, and handling to prevent unauthorized access. AES encryption performance can be optimized with hardware acceleration and parallel processing, which are supported by many modern processors. However, the security of AES depends heavily on effective key management and the use of secure modes of operation to prevent vulnerabilities in the system.

AES Security

The security of the Advanced Encryption Standard (AES) is primarily based on its use of a symmetric encryption algorithm that applies a series of rounds to transform plaintext into ciphertext. AES encryption employs key sizes of 128, 192, or 256 bits, which determine the number of rounds used in the encryption process (10 rounds for 128-bit keys, 12 rounds for 192-bit keys, and 14 rounds for 256-bit keys). AES is considered highly secure due to its large key sizes and the complexity of its

transformations, making brute-force attacks computationally infeasible. The security of AES is further enhanced by its use of multiple operations like SubBytes, ShiftRows, MixColumns, and AddRoundKey, which ensure confusion and diffusion, two principles that strengthen encryption.

6.4 Quantum Computing:

Quantum computing leverages the principles of quantum mechanics, such as superposition and entanglement, to perform computations that are infeasible for classical computers. In the context of encryption, quantum computing can significantly enhance data security by providing faster and more efficient algorithms for encryption and decryption processes. The core of quantum computing's potential lies in its ability to solve problems in parallel by utilizing quantum bits (qubits), which can represent both 0 and 1 simultaneously due to superposition. This capability allows quantum algorithms to process multiple possibilities at once, making them much faster than classical algorithms for certain tasks.

In encryption, quantum algorithms, such as Shor's Algorithm, can efficiently factorize large numbers, which threatens the security of current encryption methods like RSA. On the other hand, Grover's Algorithm can provide quadratic speedup for searching through unsorted data, which could impact symmetric-key encryption methods, including AES. However, quantum computing's potential for encryption is not limited to breaking existing cryptographic schemes—it also enables the development of quantum-resistant encryption methods.

Quantum key distribution (QKD), for example, allows for the secure sharing of cryptographic keys over potentially insecure channels, using the principles of quantum mechanics to detect eavesdropping and ensure that keys remain confidential. Moreover, quantum-safe cryptography is being developed to design encryption algorithms that are resistant to attacks by quantum computers. Quantum computing also holds promise in optimizing cryptographic operations, including data encryption in cloud computing environments. It can significantly reduce the time and resources needed for encryption, enabling the rapid encryption and decryption of large datasets, which is especially beneficial in environments with high volumes of sensitive data. As quantum computing

evolves, it is expected to complement traditional cryptographic techniques by providing more secure, efficient, and scalable solutions for data protection in the future.

Quantum Computing Security:

Quantum computing introduces new possibilities for both encryption and decryption, leveraging quantum mechanics to solve problems much faster than classical computers. The security of quantum computing, particularly in the realm of encryption, is multifaceted. While quantum computers can potentially break widely used classical encryption schemes like RSA and ECC due to their ability to efficiently solve problems such as integer factorization and discrete logarithms using algorithms like Shor's Algorithm, quantum computing also enables the creation of quantum-resistant encryption methods.

The primary security benefit of quantum computing lies in its ability to implement Quantum Key Distribution (QKD). QKD uses quantum properties such as superposition and entanglement to securely distribute cryptographic keys. Any attempt to intercept or measure the quantum key alters the quantum state, making eavesdropping detectable. This offers a high level of security, as any unauthorized interception will be immediately noticed, ensuring that only authorized parties can securely exchange keys.

6.5 Quantum-Based Encryption in ECC Algorithm:

As part of the encryption and decryption processes, this quantum-based encryption method puts to use the special capabilities of quantum computing in order to produce cryptographic keys as well as carry out encryption and decryption procedures.

6.5.1 Quantum Key Generation:

The primary idea behind this algorithm is that it is based on the use of quantum circuits to create a random cryptographic key. There is a key operation in quantum computing known as the Hadamard gate, which is essentially a superposition of states for the qubit, resulting in a more randomized outcome. Putting the Hadamard gates behind the qubits, the circuit measures their states after they have been applied to the

qubits there is an inherent randomness in quantum systems that ensures the key is unpredictable and very difficult for any adversary to replicate without access to the quantum system itself in order to obtain the key.

6.5.2 Text to Binary Conversion:

The `text_to_binary` function converts the text to be encrypted into a binary representation. Encrypting and decrypting binary data is only possible with XOR-based encryption and decryption.

def text_to_binary_v3(text):

return ' '.join(format(byte, '08b') for byte in text.encode('utf-8'))

6.5.3 Encryption: The `encrypt_text` function in the algorithm converts a bitwise XOR (exclusive OR) operation between the binary representation of the text and the quantum-generated key. XOR encryption is a simple and effective method for changing the data. In this case, the binary text is encrypted by iterating over each bit and performing XOR with the corresponding bit. If the key is shorter than the text, it wraps around and repeats the key. Encrypted code is given below:

encrypted_data = ''.join(str(int(b) ^ int(k)) for b, k in zip(binary_text, key))

This results in an encrypted binary string, which is the ciphertext.

6.5.4 Decryption: The `decrypt_text` function is reverse mechanism of encryption process by applying XOR operation again. XOR is a symmetric operation, meaning that the same key used for encryption can also decrypt the data. The encrypted binary string is XORed with the same key to recover the original binary text, which is then converted back to the original text using the `binary_to_text` function. The `decrypt_text` function is reverse mechanism of encryption process by applying XOR operation again. The decrypted is given below:

**decrypted_binary = ''.join(str(int(b) ^ int(k)) for b, k in
zip(encrypted_data, quantum_key))**

6.5.5 Advantages of Quantum Key Generation in ECC: One of the key strengths of this algorithm lies in the quantum-generated key. Traditional random number generators (RNGs) rely on algorithms and can, in theory, be predicted if an attacker has enough computational power.

CHAPTER-7

RESULT

Home page: Revolutionizing Cloud Data Security with Elliptic Curve Cryptography home page.

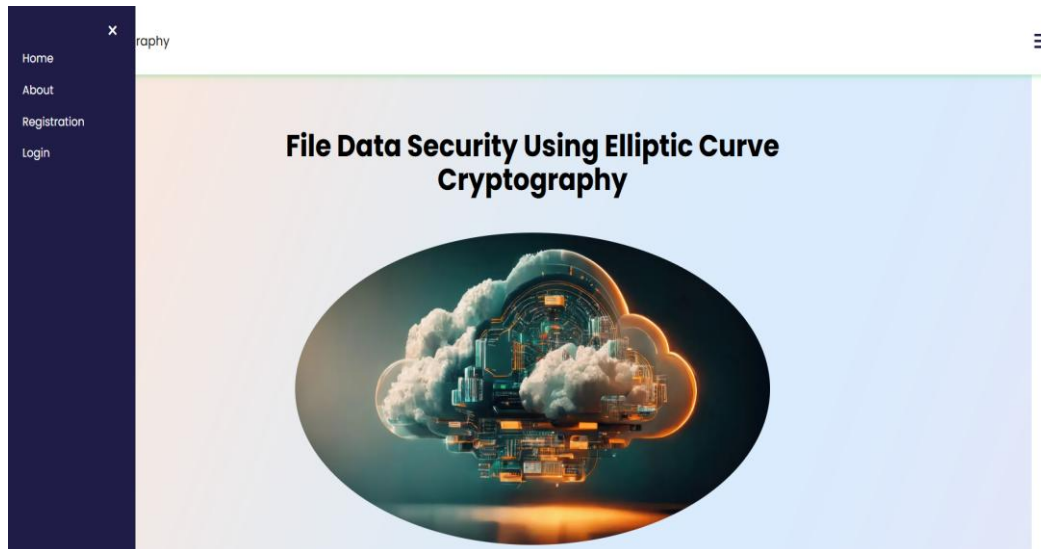


Fig. No. 7.1: Home Page

Register: Data owner, Data user and Cloud can register into website by entering details.

The image shows a web browser displaying the 'Sign Up' page. The title 'Sign Up' is centered at the top in a large, bold, black font. Below the title, there is a registration form with several input fields. At the top of the form is a dropdown menu labeled 'Data Owner'. Below this are two rows of input fields: the first row has 'Name' and 'Enter Email'; the second row has 'Password' and 'Confirm Password'. The third row has 'Contact Number' and 'Enter Address'. At the bottom of the form is a green button with the text 'REGISTER' in white capital letters.

Fig. No. 7.2: Register Page

Upload files: Here we can upload their files into website and he can encrypt that file using ECC.

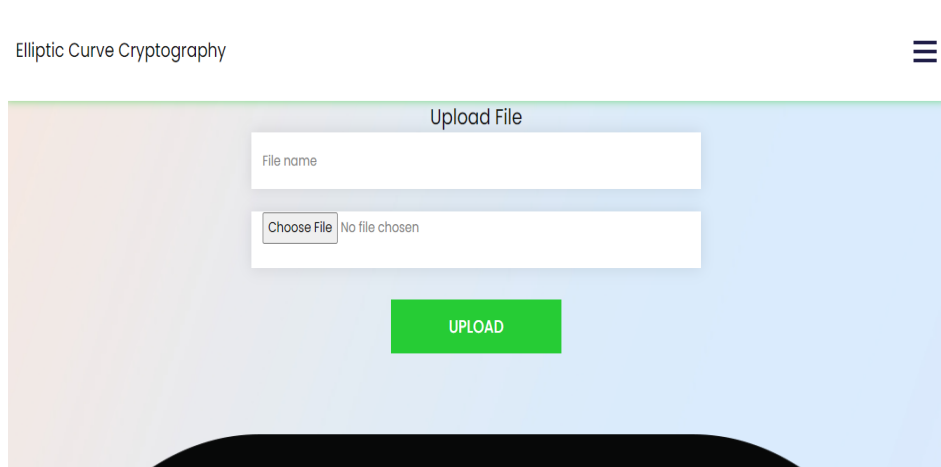


Fig. No. 7.3: Upload Files Page

Download file: We can download the file by using the key.

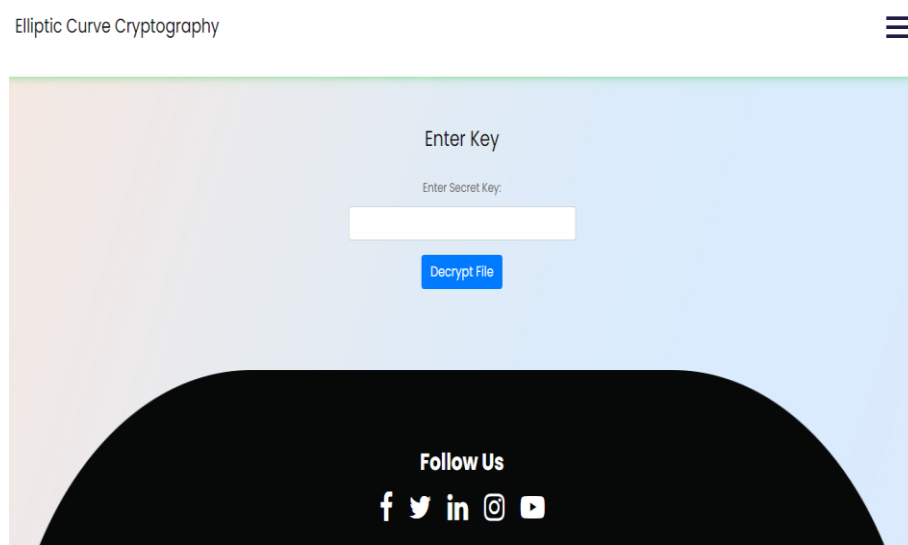


Fig. No. 7.4: Download Files Page

Graphs for both AES and ECC-Quantum:

Fig 7.5 shows AES provides the fast and efficient symmetric encryption but also faces some challenges in key management and quantum vulnerabilities, making it suitable for short-term security. As it also includes a single key for encryption and decryption.

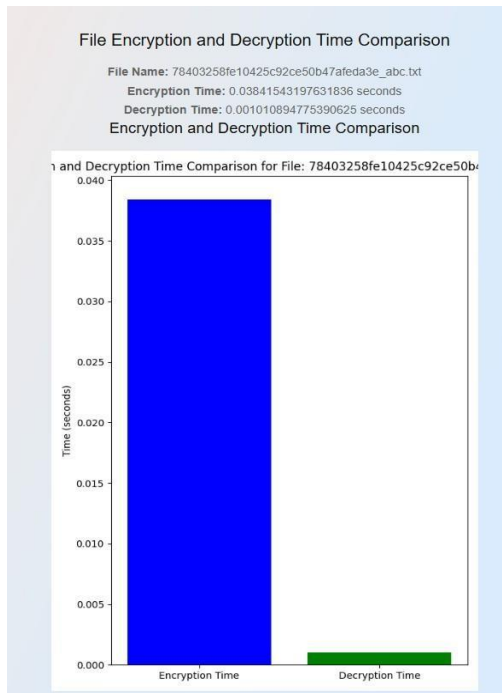


Fig. No. 7.5: AES Algorithm Graph

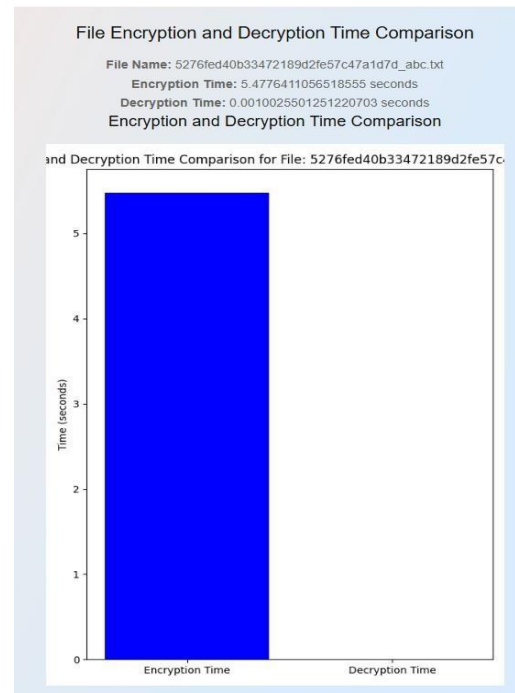


Fig. No. 7.6: ECC-Quantum Algorithm Graph

Fig 7.6 shows ECC with Quantum Cryptography provides the secure encryption, faster decryption, and reduced computational overhead. As it uses two keys for the encryption and decryption process.

CONCLUSION

In an era where data security is paramount, especially in cloud computing and IoT environments, traditional encryption methods like RSA are increasingly challenged by the need for efficiency and scalability. The proposed system, leveraging Elliptic Curve Cryptography (ECC), addresses these challenges by offering strong encryption with significantly smaller key sizes, reducing computational overhead and energy consumption. ECC's ability to deliver robust security while enhancing system performance makes it an ideal solution for modern cloud infrastructures and real-time applications. As the digital landscape continues to evolve, the adoption of ECC represents a forward-thinking approach, ensuring that data remains secure without compromising efficiency. This project underscores the potential of ECC to meet the growing demands for secure, scalable, and efficient encryption in a rapidly advancing technological world.

REFERENCES

- [1] R. Lu, X. Yuan, and X. Lin “Homomorphic Encryption for Cloud Computing: An Overview” published in IEEE Communications Surveys & Tutorials, 2021
- [2] Kyu-Seok Shim; Boseon Kim; Wonhyuk Lee “Research on Quantum Key, Distribution Key and Post-Quantum Cryptography Key Applied Protocols for Data Science and Web Security” published in IEEE Access in 2024
- [3] V. S. Pendyala, S. M. Arafath, and S. R. Kulkarni “Elliptic Curve Cryptography for Real-Time Data Encryption in IoT and Cloud Computing” IEEE Internet Of things Access, 2023
- [4] X. Kong, J. Wang, and Q. Ni "Efficient Data Security and Privacy-Preserving Scheme in Cloud Computing" in IEEE Access, 2022
- [5] IBM, "Quantum-safe cryptography: How it affects your information in the cloud," IBM Think Blog, 2024.
- [6] A. Chhabra and S. Arora “An Elliptic Curve Cryptography Based Encryption Scheme for Securing the Cloud Against Eavesdropping Attacks” by IEEE 3rd International Conference on Collaboration and Internet Computing, 2024
- [7] J. Shen, J. Niu, J. Cao, and Y. Mei, "A Survey on Cloud Security Issues and Techniques: Cryptographic and Non-Cryptographic Approaches," IEEE Transactions on Services Computing, vol. 13, no. 3, pp. 434-451, 2023.
- [8] M. S. Ali, K. K. R. Choo, and S. H. Ahmed, "Blockchain-Based Secure Data Storage and Access Control for Cloud Applications," IEEE Transactions on Cloud Computing, vol. 9, no. 3, pp. 1215-1226, 2021.
- [9] Dong Pan; Gui-Lu Long; Liuguo Yin; Yu-Bo Sheng; Dong Ruan; Soon Xin Ng; Jianhua Lu; Lajos Hanzo , ”The Evolution of Quantum Secure Direct Communication: On the Road to the Qinternet” Vol. 26 ,2024
- [10] K. Khan and R. Qazi, "Data Security in Cloud Computing Using Elliptic Curve Cryptography," International Journal of Computing and Communication Networks, vol. 1, no. 1, pp. 46-52, 2022

- [11] V. S. Miller, "Use of Elliptic Curves in Cryptography," in Conference on the Theory and Application of Cryptographic Techniques, 2020, pp. 417-426.
- [12] M. -Q. Hong, P. -Y. Wang, and W. -B. Zhao, "Homomorphic Encryption Scheme Based on Elliptic Curve Cryptography for Privacy Protection of Cloud Computing," in IEEE 2nd International Conference on Big Data Security on Cloud (BigDataSecurity), High Performance and Smart Computing (HPSC), and Intelligent Data and Security (IDS), 2023, pp. 152-157.
- [13] T. Banerjee and M. A. Hasan, "Energy Efficiency Analysis of Elliptic Curve Based Cryptosystems," in 17th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom/BigDataSE), 2018, pp. 1579-1583.



Scopus®



**International Conference on
Research and Development in Information, Communication and
Computing Technologies
(ICRDICCT'25)**

This is to certify that

K Venkatesh

**has successfully presented the paper titled
FILE DATA SECURITY USING ELLIPTIC
CURVE CRYPTOGRAPHY**

**in Proceedings of the International Conference on Research and
Development in Information, Communication and Computing Technologies
(ICRDICCT'25), held on April 4 & 5, 2025, at E.G.S. Pillay Engineering
College, Nagapattinam, Tamil Nadu, India.**

Organizing Secretary

Dr.S.Manikandan

Industry Partner

**Ms.Gugapriyaa Sivakumar
NTL Technology**

Principal

Dr.S.Ramabalan

ORGANIZED BY

**E.G.S. Pillay Engineering College,
Nagapattinam, Tamil Nadu, India.**

&

**NTL Technology
Erode, Tamil Nadu, India.**



NTL TECHNOLOGY
New Thinking and Learning

**International Conference on
Research and Development in Information, Communication and
Computing Technologies
(ICRDICCT'25)**

This is to certify that

Swaroopa Bhupalam

**has successfully presented the paper titled
FILE DATA SECURITY USING ELLIPTIC
CURVE CRYPTOGRAPHY**

**in Proceedings of the International Conference on Research and
Development in Information, Communication and Computing Technologies
(ICRDICCT'25), held on April 4 & 5, 2025, at E.G.S. Pillay Engineering
College, Nagapattinam, Tamil Nadu, India.**



Organizing Secretary

Dr.S.Manikandan



Industry Partner

**Ms.Gugapriyaa Sivakumar
NTL Technology**



Principal

Dr.S.Ramabalan

ORGANIZED BY

**E.G.S. Pillay Engineering College,
Nagapattinam, Tamil Nadu, India.**

&

**NTL Technology
Erode, Tamil Nadu, India.**



NTL TECHNOLOGY
New Thinking and Learning

**International Conference on
Research and Development in Information, Communication and
Computing Technologies
(ICRDICCT'25)**

This is to certify that

Rohini Bheemanapalli

**has successfully presented the paper titled
FILE DATA SECURITY USING ELLIPTIC
CURVE CRYPTOGRAPHY**

**in Proceedings of the International Conference on Research and
Development in Information, Communication and Computing Technologies
(ICRDICCT'25), held on April 4 & 5, 2025, at E.G.S. Pillay Engineering
College, Nagapattinam, Tamil Nadu, India.**



Organizing Secretary

Dr.S.Manikandan



Industry Partner

**Ms.Gugapriya Sivakumar
NTL Technology**



Principal

Dr.S.Ramabalan

ORGANIZED BY

**E.G.S. Pillay Engineering College,
Nagapattinam, Tamil Nadu, India.**

&

**NTL Technology
Erode, Tamil Nadu, India.**



NTL TECHNOLOGY
New Thinking and Learning

**International Conference on
Research and Development in Information, Communication and
Computing Technologies
(ICRDICCT'25)**

This is to certify that

Suvarchala Bhupalam

**has successfully presented the paper titled
FILE DATA SECURITY USING ELLIPTIC
CURVE CRYPTOGRAPHY**

**in Proceedings of the International Conference on Research and
Development in Information, Communication and Computing Technologies
(ICRDICCT'25), held on April 4 & 5, 2025, at E.G.S. Pillay Engineering
College, Nagapattinam, Tamil Nadu, India.**



Organizing Secretary

Dr.S.Manikandan



Industry Partner

**Ms.Gugapriya Sivakumar
NTL Technology**



Principal

Dr.S.Ramabalan

ORGANIZED BY

**E.G.S. Pillay Engineering College,
Nagapattinam, Tamil Nadu, India.**

&

**NTL Technology
Erode, Tamil Nadu, India.**



NTL TECHNOLOGY
New Thinking and Learning

**International Conference on
Research and Development in Information, Communication and
Computing Technologies
(ICRDICCT'25)**

This is to certify that

Yugandhar kodigi

**has successfully presented the paper titled
FILE DATA SECURITY USING ELLIPTIC
CURVE CRYPTOGRAPHY**

**in Proceedings of the International Conference on Research and
Development in Information, Communication and Computing Technologies
(ICRDICCT'25), held on April 4 & 5, 2025, at E.G.S. Pillay Engineering
College, Nagapattinam, Tamil Nadu, India.**



Organizing Secretary

Dr.S.Manikandan



Industry Partner

**Ms.Gugapriya Sivakumar
NTL Technology**



Principal

Dr.S.Ramabalan

ORGANIZED BY

**E.G.S. Pillay Engineering College,
Nagapattinam, Tamil Nadu, India.**

&

**NTL Technology
Erode, Tamil Nadu, India.**



NTL TECHNOLOGY
New Thinking and Learning