Secure Cloud Backup and Recovery System

A major project report submitted in partial fulfilment of the requirement for the award of degree of

Bachelor of Technology

in

Computer Science & Engineering / Information Technology

Submitted by

Akash Rathore(201425)

Daniel Dacosta(201190)

Under the guidance & supervision of

Dr. Nancy Singla

Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology,

Waknaghat, Solan - 173234 (India)

CERTIFICATE

This is to certify that the work which is presented in the report titled "**Secure Cloud Backup and Recovery System**" in partial fulfilment of the requirements for the award of degree of B.Tech in Computer Science and Engineering, Jaypee University of Information Technology, Waknaghat is an authentic record of work carried out by Daniel Dacosta (201190) & Akash Rathore (201425) during the period from August 2023 to May 2024 under the supervision of Dr. Nancy Singla, Department of Computer Science and Engineering, Jaypee University of Information Technology, Waknaghat.

Student Name: Daniel Dacosta	Student Name: Akash Rathore
Roll No.: 201190	Roll No.: 201425

The above statement made is correct to the best of my knowledge.

Supervisor Name: Dr. Nancy Singla
Designation: Assistant Professor(SG)
Department: Department of Computer Science & Engineering and Information Technology
Dated:

Candidate's Declaration

We hereby declare that the work presented in this report entitled **'Secure Cloud Backup and Recovery System'** in partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science & Engineering / Information Technology** submitted in the Department of Computer Science & Engineering and Information Technology, Jaypee University of Information Technology, Waknaghat is an authentic record of my own work carried out over a period from August 2023 to May 2024 under the supervision of **Dr. Nancy Singla** (Assistant Professor(SG), Department of Computer Science & Engineering and Information Technology).

The matter embodied in the report has not been submitted for the award of any other degree or diploma.

Student Name: Daniel Dacosta
Roll No.: 201190

Student Name: Akash Rathore
Roll No.: 201425

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

Supervisor Name: Dr. Nancy Singla
Designation: Assistant Professor(SG)
Department: Department of Computer Science & Engineering and Information Technology
Dated:

ACKNOWLEDGEMENT

Firstly, we express our heartiest thanks and gratefulness to almighty God for His divine blessing makes it possible to complete the project work successfully.

We are really grateful and wish our profound indebtedness to Supervisor Dr. Nancy Singla, Assistant Professor(SG), Department of CSE Jaypee University of Information Technology,Wakhnaghat. Deep Knowledge & keen interest of my supervisor in the field of Cloud and Information Security to carry out this project. Her endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stages have made it possible to complete this project.

We would like to express our heartiest gratitude to Dr. Nancy Singla, Department of CSE, for her kind help to finish our project.

We would also generously welcome each one of those individuals who have helped us straightforwardly or in a roundabout way in making this project a win. In this unique situation, we might want to thank the various staff individuals, both educating and non-instructing, which have developed their convenient help and facilitated my undertaking.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

Akash Rathore	Daniel Dacosta
(201425)	(201190)

TABLE OF CONTENTS

LIST OF TABLE

LIST OF FIGURES

LIST OF ABBREVIATIONS

AES	Advanced Encryption Standard
RSA	Rivesh, Sharim, Adleman
AWS	Amazon Web Services
GDPR	General Data Protection Regulation
PaaS	Platform as a Service
SLA	Service Level Agreement
HTTP	Hypertext Transfer Protocol
OTP	One Time Password
API	Application Programming Interface
TLS	Transport Layer Security
SDK	Software Development Kit

ABSTRACT

In today's virtual environment, the want for reliable records garage, protection and restoration mechanisms is paramount. “Secure Cloud Backup and Recovery System" represents an entire answer designed to fulfil these critical wishes. This project aims to provide a strong and stable platform for records backup and healing in cloud environments, prioritising information integrity, confidentiality and accessibility.

The main motive of this project is to provide users with an unbroken and dependable way to guard their facts towards possible loss, corruption or unauthorised access. Using superior encryption protocols and authentication mechanisms, the device guarantees that sensitive records are covered in the cloud infrastructure each in transit and at relaxation.

In addition, the project emphasises efficiency and scalability, permitting seamless growth of garage capability at the same time as maintaining most effective performance. Using dispensed storage technology and redundant backups, it minimises the chance of records loss because of hardware failure or surprising occasions, improving standard reliability.

A crucial function of steady cloud backup and restoration is its consumer-friendly interface that permits users to without problems initiate and control backups, agenda automatic backups, and fast restore information when wished. This consumer-centric method aims to reduce downtime and streamline the restoration technique, making sure commercial enterprise continuity and continuity.

The project and its structure encompass a multi-layered protection framework that combines exceptional layers of authentication, access control and encryption strategies. This multi-layered method hardens the device in opposition to ability vulnerabilities, lowering the likelihood of statistics breaches and unauthorised get right of entry to.

In brief, it may be said that the Secure Cloud Backup and Recovery System is a key solution inside the field of information management and protection. Combining today's safety features with consumer-pleasant capability, it gives a strong, scalable and stable surroundings for records backup and recuperation in cloud-primarily based infrastructure.

CHAPTER 1: INTRODUCTION

1.1 INTRODUCTION

In a technology defined by way of the usage of the explosion of digital information, statistics retention and protection are paramount for each person and businesses. The creation of cloud generation has revolutionised facts control, presenting unheard of flexibility and accessibility.

However, the transition to cloud-based total storage requires sturdy mechanisms to make sure information integrity, confidentiality and recoverability from capability threats and vulnerabilities.

"Secure Cloud Backup and Recovery System" emerges as a central answer adapted to the evolving demanding situations of file protection and healing in cloud environments. This revolutionary system represents a fusion of modern generation and careful layout to provide a stable, green and scalable platform to guard essential statistics.

Our project is basically designed to reduce the risks related to records loss, corruption or unauthorised right of entry by using advanced encryption protocols, multifactor authentication and complete entry to controls. By implementing these protection functions, it guarantees that facts are covered from outside threats sooner or later of its lifecycle within the cloud infrastructure.

In addition, the "Secure Cloud Backup and Recovery System" prioritises unfastened operation and accessibility, permitting customers to effect listing, adjust timing and quickly restore information while wanted. This man or woman-targeted method not most effective increases the general performance of the operation, however additionally reduces the impact of capability interruptions, facilitating speedy records development and organisational continuity.

The format of this challenge is cautiously deliberate and includes redundant backups, isolated cloud storage mechanisms and fault-tolerant protocols that harden in opposition to hardware failures or sudden operations. This redundancy and fault tolerance minimises potential statistics loss and downtime, making sure uninterrupted get right of entry to key records.

In the midst of an ever-evolving cyber hazard, the "Secure Cloud Backup and Recovery System" is a testimony to innovation and resilience. Combining ultra-modern protection protocols with intuitive user interfaces, this system redefines the facts, safety paradigm, making sure peace of thoughts and commercial enterprise continuity in a technology in which statistics integrity is paramount.

This project sets the degree to discover the complex layers of security, performance and reliability of the Secure Cloud Backup and Recovery system. As we dig deeper, we reveal its flexible layout, functionality and relevant role in protective and recuperating data from cloud-based infrastructures.

1.2 PROBLEM STATEMENT

Nowadays in the information environment, there is a simple need for a reliable and stable cloud-based backup and restoration device. As the protection of important facts is predicated closely on cloud offerings, there may be an urgent need for a device that guarantees uncompromising information protection, integrity and easy recuperation in the event of ability threats or device disasters.

Although cloud storage gives convenience, it increases severe issues approximately the safety of sensitive facts stored on far flung servers. Traditional backup techniques frequently lack critical encryption and confidentiality additives, leaving information susceptible to fact breaches, unauthorised access and inadvertent disclosure. In addition, the lack of sturdy healing mechanisms ends in long downtimes and malfunctions all through gadget failures.

To solve those urgent troubles, it is important to develop a sophisticated encrypted cloud backup and recovery system. This project should seamlessly combine cease-to-stop encryption, consumer-friendly interfaces, effective backup techniques and flexible recuperation options. It has to conquer the limitations of current solutions by enforcing effective encryption strategies, rigorous access control and modern restoration techniques to fulfil numerous consumers wishes while complying with privacy policies and enterprise requirements.

Successful implementation of this project calls for a multidisciplinary technique that combines cryptography, cloud technology, software program and user revel in. The aim of the resulting project is to instil confidence in users and ensure the confidentiality of their data and their retrieval, even in the event of surprising and demanding situations.

In addition to security and healing concerns, the ultra-modern world lacks an unmarried solution that seamlessly integrates encryption, personal accessibility and scalability with cloud-based backup structures. Current methods frequently compromise either comfort or strong security with the aid of now not imparting a single platform that balances strict privacy measures with person-friendly interfaces for exclusive file storage needs.

Encrypted cloud backup and recovery isn't the simplest technological solution; however, it contributes to the development of records safety, privacy and control. Directly addressing these challenges, this initiative aims to redefine how information is covered, secured and recovered in our increasing number of interconnected virtual panoramas.

1.3 OBJECTIVES

To create “Secure Cloud Backup and Recovery System”, the following goals describe the important milestones and obligations required for its a hit improvement and implementation.

- **ANALYSING EXISTING SYSTEMS:** Evaluate and assess common cloud backup and restoration systems to discover their strengths and weaknesses and the way they perform. This evaluation is the idea for figuring out gaps and opportunities for improvement.

- **RESEARCHING SECURITY MEASURES:** Explore and examine encryption strategy, authentication protocols, and first-class practices for a secure cloud infrastructure. This file offers statistics on selecting the most effective security features and machine integration strategy to provide an efficient modern solution.

- **DESIGNING COMPREHENSIVE SYSTEMS ARCHITECTURE:** Develop a care plan that includes a unified system architecture. This architecture harmoniously combines end-to-end encryption, a strong user authentication mechanism and the use of secure cloud resources to provide a stronger privacy environment.

- **PLANNING USER INTERFACE AND RECOVERY STRATEGIES:** Design the look and functionality of the user interface, emphasising user accessibility without compromising security. In addition, various recovery options and disaster recovery strategies are available to ensure fast data recovery in the event of unexpected events.

- **IMPLEMENT SYSTEM FRAMEWORKS:** Translate the proposed architecture into useful code with a focus on growing the logical framework for person interfaces and data serving. This section involves coding and programming based on the provided architectural specifications.

- **CONDUCT RIGOROUS TESTING:** Run a complete trying out software that includes functionality critiques, rigorous protection critiques, and person revel in exams. This step guarantees machine and system reliability, resilience against capacity threats and foremost availability before deployment.

Together, those goals define a systematic approach to developing a consistent cloud backup and recuperation machine that guarantees advanced records integrity, guarantees stable get entry to and allows seamless healing in cloud environments.

1.4 SIGNIFICANCE AND MOTIVATION OF OUR PROJECT

Nowadays in virtual surroundings, the significance of a stable cloud backup and restoration machine is going past mere convenience; it's miles the cornerstone of records integrity, privateness and business organisation continuity. The motivation to make any such device bigger is deeply rooted in statistics safety, statistics breaches, and addressing the vital need for fast and solid information recovery mechanisms in cloud-based total environments.

1.4.1 SIGNIFICANCE

- **DATA SECURITY ASSURANCE:** With evolution of cyber threats, the importance of defensive sensitive statistics cannot be overruled. The Secure Cloud Backup and Recovery System strives to encourage endure in mind along implementing strong encryption, authentication measures and strong cloud infrastructure practices, ensuring that facts are blanketed from unauthorised get right of entry to and information breaches.

- **OPERATIONAL CONTINUITY:** The strolling of agencies and businesses is depending on the undisturbed availability of crucial data. This project and its importance lie in its capacity to offer quick and reliable healing abilities, lowering downtime inside the route of recorded situations and improving organization continuity.

- **COMPLIANCE WITH REGULATIONS:** Compliance with strict statistics safety rules and industry requirements is crucial. The machine and improvement comply with those guidelines, ensuring compliance with statistics protection legal guidelines and protecting in opposition to possible consequences for violations.

- **ENHANCED USER CONFIDENCE:** Enhanced User Confidence: Giving users an advanced user-friendly interface, the project increases confidence in cloud storage. To encourage the use of cloud technologies in various fields there should be an increase in innovation and efficiency.

1.4.2 MOTIVATION

- **ADDRESSING EXISTING VULNERABILITIES:** The motivation comes from reputation of the vulnerabilities and obstacles of current cloud backup and restoration structures. By bridging those gaps, this task aims to provide a greater strong and dependable answer that meets the converting goals of data safety.

- **INNOVATION AND TECHNOLOGICAL ADVANCEMENT:** The motivation of the challenge is to push the bounds of technological innovation. It strives to apply brand new encryption strategies, authentication mechanisms and cloud infrastructure practices to set new benchmarks for statistics protection and recuperation.

- **EMPOWERING USERS:** Driven via the selection to empower people and organisations, this mission aims to offer a customer-centric device that puts data management and recovery competencies right away in the palms of users, improving their records management.

- **CONTRIBUTING TO THE FIELD:** In addition to performing as an answer, the inducement is to contribute to protection and records healing. The project aims to provide information, first-class practices and methods that can be useful to the wider community and increase the sphere of stable cloud technology.

Basically, the importance and motivation of the Secure Cloud Backup and Recovery System lies in its ability to reinforce information security, ensure the sustainability of operations, follow regulations, instil the trust of users and sell the continuous improvement of secure cloud technology.

1.5 ORGANISATION OF PROJECT REPORT

CHAPTER 1 - INTRODUCTION: This chapter serves as a project and launching pad to provide a comprehensive background for secure cloud backup and recovery. It includes an introduction to the project, defines the problem, outlines the objectives, reflects the purpose and motivation of the project and decides the organisation of the project preview report.

CHAPTER 2 - LITERATURE SURVEY: This chapter focuses on extensive research and integrates information from a variety of reputable sources, including standard books, journals, websites, and technical publications. The pertinent literature is printed, prior research on the subject is highlighted, and vital traits in cloud storage, backup, healing, and encryption are mentioned.

CHAPTER 3 - SYSTEM DEVELOPMENT: This chapter discusses the technical additives in the order of requirements and evaluation, venture planning, and architecture. It covers the statistics education and implementation system and introduces the most vital parts which include code samples, algorithms, equipment and strategies. The fundamental difficulties of the development procedure and the way to remedy them are also examined.

CHAPTER 4 - TESTING: This chapter discusses the trying out technique, emphasising the devices and strategies used to evaluate the platform and its performance, emphasising the platform and ensuring reliability. It presents a thorough evaluation of the system with the aid of supplying a summary of take a look at instances and their outcomes.

CHAPTER 5 - RESULTS AND EVALUATION: Presenting and analysing the project and results is our main goal at this stage. Where applicable, it provides platform and performance benchmarks, providing a comparison with current solutions and a comprehensive presentation of the results and their interpretation.

CHAPTER 6 - CONCLUSIONS AND FUTURE SCOPE: This insightful chapter provides an overview of the project and key findings, limitations and industry contributions. In summary, the future opportunities and potential opportunities for progress and growth in the field of secure cloud technology are outlined. It provides an early insight into the platform and its development.

CHAPTER 2: LITERATURE SURVEY

2.1 OVERVIEW OF RELEVANT LITERATURE

A complete assessment of latest studies highlights several strategies and strategies in stable cloud backup and recuperation structures. In these studies, there is an apparent try to enhance information protection, integrity and rapid restoration mechanisms in cloud environments. Several key problems emerge, highlighting each of the advances and barriers of modern-day techniques.

- **ENCRYPTION AND SECURE MONITORING:** Several articles together with "A secure database monitoring method to improve data backup and recovery operations in cloud services"[1] and "A Dynamic Four-Step Data Security Model for Data in Cloud Computing Based on Cryptography and Steganography"[2] emphasizes the importance of encryption and tracking in strengthening backup and recuperation operations. Although the techniques have improved the safety and scalability of backup tempo, empirical effects and answers to implementation troubles are lacking.

- **COSTS AND SCALABILITY:** Studies like "Cloud Computing Storage Backup and Recovery Strategy Based on Secure IoT and Spark"[3] and "Disaster Recovery Techniques in Cloud Computing"[7] delve into the garage, conversation and scalability expenses of cloud backup techniques. They spotlight how verbal exchange costs grow with record duration and pattern ratio, affecting reaction instances and question prices, despite the fact that they no longer very well investigate technical elements and reliability issues.

- **MULTI-CLOUD AND HYBRID APPROACHES:** "DropStore: A Secure Backup System Using Multi-Cloud and Fog Computing"[4] examines the effectiveness of several cloud and fog computing strategies to reap stable backups. Despite highlighting the ability advantages of this technique, the paper lacks technical depth, evaluation and dialogue of statistics reliability.

- **SECURITY CONTROLS AND RISK ASSESSMENTS:** The literature additionally consists of research which include "CyberSecurity Architecture for the Cloud: Protecting Network in a Virtual Environment"[5] and "Secure Cloud Computing: Benefits, Risks and Controls,"[11] that specialize in safety controls, danger assessment frameworks and the importance of compliance in cloud environments. However, these studies from time to time lack unique effects on cloud compatibility and boundaries to cloud frameworks.

- **PERFORMANCE TRADE-OFF AND DEDUPLICATION**: Research articles together with "Tapping the Potential: Secure Chunk-based totally Deduplication of Encrypted Data for Cloud Backup"[8] and "Overview of information backup and catastrophe restoration in cloud"[10] highlights the alternate-offs between security, redundancy and overall performance. While these techniques sacrifice redundancy for higher protection and quicker information recuperation, moreover they introduce boundaries on scalability and compatibility.

These researches shed light on various factors of stable cloud backup and recuperation structures, highlighting the significance of encryption, monitoring, price concerns, multi-cloud approaches, protection and overall performance. However, the literature additionally highlights empirical proof, implementation challenges, scalability debates and complete evaluations, suggesting opportunities for destiny research and development in this critical region.

Table 2.1 : Literature review

S. No.	**Paper Title [Cite]**	**Journal/ Conference (Year)**	**Tools/ Techniques/ Dataset**	**Results**	**Limitations**
1.	A Secured Database Monitoring	BOHR International Journal of Computer	Encryption, Secured Database Monitoring Method	Database Monitoring Method enhances cloud data	Absence of empirical results or case studies. Fails to

	Method to Improve Data Backup and Recovery Operations in Cloud Computing.[1]	Science (2023)		backup and recovery operations. Backup speed scales with data volume.	address potential challenges and limitations in implementation.
2.	A Dynamic Four-Step Data Security Model for Data in Cloud Computing Based on Cryptography and Steganography.[2]	Sensors (2022)	AES, RSA Google Drive	Outlines cloud data security model: encryption, steganography, backup, sharing, Enhancing privacy.	unauthorised access by cloud administrators. No specific mention of data sharing using the proposed security model.
3.	Cloud Computing Storage Backup and Recovery Strategy Based on Secure IoT and Spark.[3]	Hindawi Mobile Information Systems (2021)	Oppositional Group Search Optimizer Algorithm, Titian Library, Apache Spark	Communication costs rise with file size and sample ratio. Response and inquiry costs are comparable.	Calculation costs grow with file size and sample ratio. Costs rise with an increase in the sample ratio.

4.	DropStore: A Secure Backup System Using Multi-Cloud and Fog Computing. [4]	IEEE Access (2021)	Fog Computing, Secure File Transfer Protocol.	Secure backup with multi-cloud, encryption, and performance considerations.	Document lacks technical depth, evaluation, comparisons, and data reliability.
5.	CyberSecurity Architecture for the Cloud: Protecting Network in a Virtual Environment. [5]	International Journal of Intelligent Automation and Computing (2021)	Distributed Denial of Services Mitigation Framework Network, Processing as a Cloud Service.	Key components: risk, training, and incident response. Extensive security controls to assess cloud service security.	Lacks specific information about the outcomes of cloud compliance. No mention of the hurdles of cloud framework.
6.	Cloud Backup & Recovery Techniques of Cloud Computing. [6]	International Research Journal of Engineering and Technology (2020)	Parity Cloud Service, Shared Backup Router Resources Architecture.	AWS and Azure both have strong features. AWS stands out for flexibility.	Azure is comparatively harder to use and manage. Azure is more expensive.

7.	Disaster Recovery Techniques in Cloud Computing. [7]	IEEE Xplore (2019)	Linux Box Parity Cloud Service High Security Distribution and Rake Technology Efficient Routing Grounded on Taxonomy.	Main cloud server in compression results is reduced, backup server memory needs. Calculate disaster recovery cost in app.	Cloud computing faces liability challenges for illegal data, including compliance issues.
8.	Tapping the Potential: Secure Chunk-based Deduplication of Encrypted Data for Cloud Backup. [8]	IEEE Conference on Communications and Network Security (2018)	Content-aware deduplication, Random Oracle Key Generation protocol.	Sacrifices deduplication for improved security, faster data restore. Enables randomised key generation.	Rate-limiting strategy compatibility, Compatibility with public cloud backup service.
9.	Towards a Big Data system disaster recovery in a Private Cloud. [9]	Ad Hoc Networks (2015)	Network Attached Storage, VMware.	A multi-technique approach achieving 99.9% data recovery for secure backup and retrieval.	No details on transferring over 3.6 TB files to other VMs. Missing scalability discussion.

10.	Overview of data backup and disaster recovery in cloud. [10]	ICICES (2014)	High-Security Distribution and Rake Technology Linux Box Cold and Hot Backup Service Replacement Strategy.	Overview of cloud-based data backup & recovery. Emphasis on business and impact evaluation.	Recognition of the limited uses of cloud-based computing and its vulnerability to specific threats.
11.	Secure cloud computing: Benefits, risks and controls. [11]	IEEE ICPPW (2011)	PaaS SLA Virtualization	Guidelines for managing cloud security risks. Information security as the top risk in cloud and virtualization.	Security risks, less on cost efficiency, scalability. Lacks specific quantitative data on risk occurrences in cloud computing.
12.	A Secure Cloud Backup System with Assured Deletion and Version Control. [12]	IEEE ICPPW (2011)	AES, Encrypted De-duplication Revocable Backup System.	Dataset statistics include file count, median, average, maximum, and total snapshot sizes on the first and last day.	No explicit limitations mentioned for FadeVersion or the conducted experiments in the document.

13.	Ensure Data Security in Cloud Storage. [26]	International Conference on Network Computing and Information Security (2011)	SLA, IaaS, AWS EC2	Framework to ensure the data security in cloud storage systems.	Lacks detailed information on the effectiveness of the proposed system.
14.	Ensuring Data Storage Security in Cloud Computing. [27]	17th International Workshop on Quality of Service (2009)	Homomorphic Tags Erasure-Correcting Code Challenge-Response Protocol Reed-Solomon Coding	Uses homomorphic tokens and distributed verification of erasure-coded data to guarantee storage accuracy and localize data errors	A probability component in the process even if successful file retrieval is quite likely with the right settings and sufficient verifications.

2.2 KEY GAPS IN THE LITERATURE

Examining the frame of studies on secure cloud backup and recuperation systems exhibits some of giant holes that want to be filled.

- **LACK OF EMPIRICAL RESULTS OR CASE STUDIES**: The first paper presents a method of monitoring the database through a smart mechanism which allows the data backup and recovery operations to be improved in the cloud computing. Although the technique is specified, it cannot present the experimental data or the cases of its successful application in actual life situations. The absence of empirical evidence makes it hard to understand the real-life applications and the productivity of the suggested method.

- **ABSENCE OF ADDRESSING POTENTIAL CHALLENGES AND LIMITATIONS:** Although the second paper provides a compelling four-step data security model for cloud computing based on cryptography and steganography, it does not explore the possible issues that may occur during the execution of the proposed model. The problem can be seen by the difficulties in developing the suitable models to deal with those problems. The understanding of these problems is essential for the design of the strategies which will reduce the risks and make the model feasible in any environment.

- **LACK OF TECHNICAL DEPTH, EVALUATION, COMPARISONS, AND DATA RELIABILITY:** DropStore, which is a reliable backup system using the combination of multi-cloud and fog computing, is the subject of the fourth paper. On the other hand, it is not detailed enough, it does not offer a full evaluation, it does not compare it with other existing solutions and it does not verify the data's reliability. By not doing the thorough evaluation and comparison with the other methods that have been proven to work, it is hard to tell about the effectiveness and reliability of DropStore as a reliable backup solution.

- **INSUFFICIENT INFORMATION ABOUT OUTCOMES AND HURDLES:** Through the fifth paper, the author proposes a cybersecurity architecture for the cloud environments by stressing the risk assessment, training, and incident response. Nevertheless, it does not contain any detailed information about the results of the cloud compliance activities and it does not even touch upon the difficulties that may arise while implementing the proposed architecture. The first thing to be done is to comprehend the results and the difficulties and this is vital to the improvement of the cybersecurity architecture so that it can be used as a tool to protect cloud networks.

- **MISSING SCALABILITY DISCUSSION AND DETAILS ON FILE TRANSFER:** The ninth paper presents the disaster recovery methods in cloud computing but doesn't mention the scalability and also, it doesn't explain how to transfer the large files to other virtual machines. Scalability is the key element of the growing data volumes that will be able to be accommodated as the time goes by, while information on file transfer is the necessary part for the understanding of the practical meaning of the proposed techniques in real life scenarios.

- **LIMITED QUANTITATIVE DATA ON RISK OCCURRENCES:** The eleventh paper talks about the advantages, dangers, and regulations of secure cloud computing, but it does not have specific quantum data on the risk occurrences of cloud environments. Quantitative data on the risk occurrences would be a guide for the organizations to identify the frequency and the severity of the security threats and thus, they will be able to focus on the areas that need the most attention and allocate the resources accordingly.

CHAPTER 3: SYSTEM DEVELOPMENT

3.1 REQUIREMENTS AND ANALYSIS

In the improvement of the Secure Cloud Backup and Recovery System, a comprehensive evaluation of requirements paperwork the muse for the following design and implementation levels. This section outlines the essential useful and non-useful requirements, together with the evaluation undertaken to derive the ones specifications.

3.1.1 FUNCTIONAL REQUIREMENTS

1. **FILE UPLOAD:** Without requiring an account or registration, users should be able to upload a variety of items, including documents, photos, videos, and archives, to the system. All file uploads, regardless of size or format, should be supported by the system without any issues.

   To avoid misuse or resource depletion, the system ought to impose a programmable maximum file size restriction for individual file uploads. Any size limitations should be made known to users before they begin the upload procedure.

   With simple drag-and-drop capability or file selection dialogues, an intuitive interface should make it simple to choose and upload files. Real-time feedback on the upload process, including upload speed and remaining time, should be provided by progress indicators.

   Before being processed further, uploaded files should be checked for viruses or malware. For the purpose of performing real-time file scanning and threat detection, the system should be integrated with antivirus scanning tools or APIs.

2. **EMAIL VERIFICATION:** Email Validation: Users should be asked to input the email address connected to the submitted file after a successful file upload. The email address format should be verified by the system, and its uniqueness should be guaranteed.

The system should use a reputable email delivery provider (like SMTP) to create a one-of-a-kind, time-limited OTP and send it to the specified email address. To avoid manipulation or prediction, the OTP should be securely created using cryptographic techniques.

By entering the obtained OTP within the allotted period, users should confirm that they are the owner of the submitted file and that their email address. Brute-force efforts and replay attacks should not be able to penetrate the OTP verification procedure.

Processing of the file should only proceed including encryption and storage after the email verification procedure has been successful. Users should be alerted and asked to redo the verification procedure in the event that it fails or expires.

3. **FILE ENCRYPTION:** The uploaded file should be encrypted using AES-CBC using a randomly generated 256-bit key when the email verification process is completed. To guarantee unpredictability and randomness, the encryption procedure should be carried out using a cryptographically secure pseudorandom number generator (CSPRNG).

   The RSA public key encryption technique should be used to encrypt the AES key, using a strong key length (e.g., 2048 bits or more). Padding techniques should be used with the RSA encryption process to improve security and thwart known cryptographic attacks.

   It is recommended to develop secure key creation, management, and storage procedures to guard against unauthorised access or manipulation. Key management protocols, appropriate entropy sources, and well-established cryptographic libraries or modules should all be used for generating keys.

4. **KEY MANAGEMENT:** Every file that a user uploads needs to be linked to a distinct pair of RSA keys (public and private keys). Secure key pair generation should be accomplished with industry-standard cryptographic parameters and techniques.

   Each uploaded file's private key has to be safely sent to the user by email or a secure download link. To avoid interception or eavesdropping, transmission channels should be encrypted using secure transport protocols (such as TLS).

Together with file metadata, every file's public key has to be safely kept in the system's database. Public keys should be secured using encryption and access restrictions to prevent unauthorised access or alteration.

To prevent unauthorised access or disclosure of the public keys that are saved, secure key storage procedures (such as encryption at rest and access restrictions) should be followed.

5. **FILE STORAGE:** To save storage space and increase transmission rates, the encrypted file should be compressed using a lossless compression technique (such as Gzip or Bzip2). To reduce processing overhead, compression should be carried out utilising effective libraries or modules.

   It is important to provide secure storage in a cloud storage service (such as Google Cloud Storage, Amazon S3) with suitable access restrictions and encryption at rest. To safely store encrypted data in specified storage buckets or containers, the system should interface with cloud storage APIs or SDKs.

   The system database should contain metadata for every file that is saved, such as the owner's email address, encryption information, and other pertinent data. To make file retrieval and management processes more efficient, metadata ought to be searchable and indexed.

6. **FILE MANAGEMENT:** Users should be able to see a list of the files they have uploaded, including with any pertinent metadata (such as the file name, size, and upload date). There should be options for pagination, sorting, and filtering the file list in an easy-to-use manner.

   It should be necessary to use two-factor authentication—a private key upload and an email OTP verification—when downloading or deleting files. Before being able to use file management features, users should be sent OTPs by email and asked to input them in order to authenticate themselves.

   The system should download the encrypted file from cloud storage, use the user's private key to decrypt it, and then provide the user the file that has been decrypted. Secure decryption with well-known cryptographic libraries or modules is recommended to avoid unwanted access to private information.

The encrypted file should be safely deleted from cloud storage, and metadata changes ought to take place at the same time. For the purpose of permanently removing data from storage and preventing unauthorised parties from accessing them, the system needs to have secure deletion techniques.

7. **ERROR HANDLING AND LOGGING:** During file upload, encryption, storage, and management activities, a variety of error scenarios should be handled gracefully via strong error handling systems. Logging and classifying errors in a methodical manner will help in troubleshooting and solving them.

   Secure, tamper-evident logging should be kept for all important actions (such as uploads, downloads, deletions, and authentication events). To facilitate auditing, monitoring, and forensic analysis, logs should contain specific data such as timestamps, user IDs, and operation results.

   Users should get error messages that are instructive and practical, with precise instructions on how to fix problems that arise. Error messages shouldn't divulge private information or reveal implementation specifics that an attacker may use against.

3.1.2 NON-FUNCTIONAL REQUIREMENTS

1. **PERFORMANCE:** Response Time: The system should respond quickly to user inputs. File uploads, email verifications, and downloads should all be completed in an acceptable amount of time, depending on the size of the file (for example, files up to 1 GB should be handled in 2 minutes).

   Throughput: Without experiencing any performance deterioration, the system should be able to manage a large number of concurrent requests and support at least 1000 file uploads and downloads every hour.

   Scalability: The system design should be able to grow horizontally by adding more server instances or cloud resources as needed, in order to handle rising user traffic and data volumes.

2. **SECURITY:** Data Encryption: To prevent unauthorised access, all sensitive data, including uploaded files and encryption keys, should be encrypted using robust cryptographic techniques (such as RSA-2048 and AES-256).

   Access Control: Based on user roles and permissions, role-based access control (RBAC) mechanisms should be put in place to limit access to sensitive functionality and data.

   Authentication & Authorization: Before allowing access to file download and deletion actions, users' identities should be confirmed through the use of secure authentication procedures, such as multi-factor authentication (MFA) that employs private keys and one-time passwords.

3. **RELIABILITY:** High Availability: To guarantee that users can use the system without interruption, the system should maintain high availability, with uptime objectives of at least 99.9%.

   Data Integrity: To identify and stop data corruption or tampering during file uploads, downloads, and storage activities, strong data integrity procedures, such as data validation checks, should be put in place.

   Backup and Recovery: To reduce the chance of data loss and enable prompt recovery in the case of system failures or disasters, regular data backups and disaster recovery procedures should be in place.

4. **USABILITY:** User Interface Design: For file upload, email verification, and download/deletion procedures, the user interface should include clear and easy-to-use navigation pathways, a consistent layout, and helpful feedback messages.

   Accessibility: To guarantee that users with disabilities can access and engage with the system efficiently, the system should adhere to accessibility standards, such as Web Content Accessibility Guidelines.

5. **SCALABILITY:** Horizontal Scaling: In order to efficiently distribute workload and manage growing traffic, the system design should allow for the addition of more server instances or cloud resources.

Fault Tolerance: To guarantee continuous service availability and data integrity in the case of hardware failures or service outages, redundancy mechanisms, such as data replication across several availability zones or regions in the cloud storage service, should be put into place.

3.1.3 ANALYSIS APPROACH

To extract functional requirements, use cases, user stories, and workflow diagrams were used. Risk analyses and performance testing simulations were used to identify security and performance considerations.

1. **REQUIREMENT PRIORITIZATION:** To get functional requirements, user stories, and use cases, employ strategies like surveys and brainstorming. To see how users interact with the system and its features, draw basic process diagrams. Sort needs into priority lists according to their significance, viability, and compatibility with the project's resources and schedule.

2. **RISK ANALYSIS AND PERFORMANCE CONSIDERATIONS:** To find possible security concerns and performance bottlenecks, undertake a basic risk analysis. Conduct basic simulations for performance testing, such load testing using simulated user traffic, to evaluate the responsiveness of the system in common usage situations. Within the parameters of the project, take into account security methods such as user authentication and data encryption while keeping an emphasis on basic concepts.

3.2 PROJECT DESIGN AND ARCHITECTURE

The design and architecture of the Secure Cloud Backup and Recovery System are essential parts of the development process. This part presents an in-depth description of the chosen architectural components, system modules, and the reasons for their integration.

3.2.1 SYSTEM ARCHITECTURE

The Secure Cloud Backup and Restore System's architecture and design are essential to guaranteeing its scalability, security, and usefulness. An overview of the architectural design is given in this part along with a detailed examination of the tools, languages, and technologies used in the system's architecture.

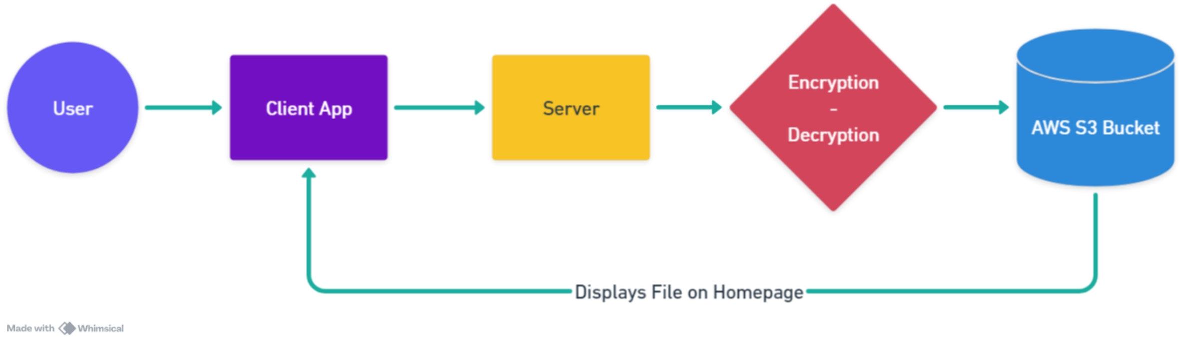
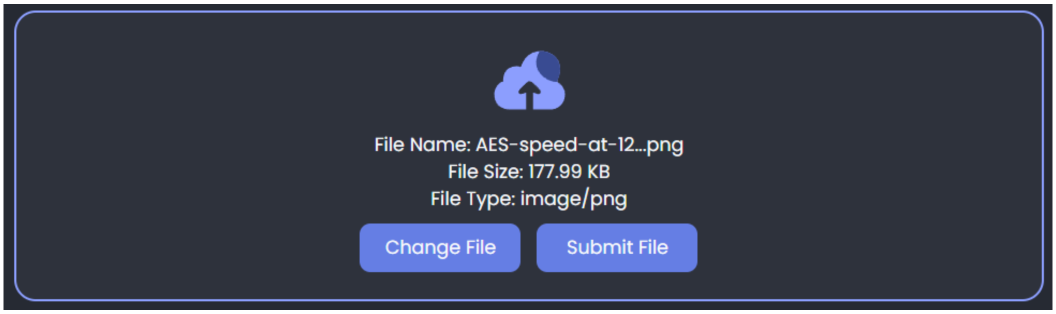


Fig. 3.1 : User flow of the Application

Architecture of the System

For backup and recovery operations, security, scalability, and efficiency are given top priority in the carefully designed system architecture. Let's examine the main elements and their functions in more detail:

Interface User: The main point of contact between users and the system is the user interface. React.js with Tailwind CSS enable the development of a visually beautiful and responsive UI. Component-based UI development is made easier by React.js, which also makes state management and modular design possible. Tailwind CSS serves as a counterbalance by prioritizing usefulness above style, guaranteeing uniformity and adaptability across various screen sizes and gadgets.



File Name: AES-speed-at-12...png
File Size: 177.99 KB
File Type: image/png

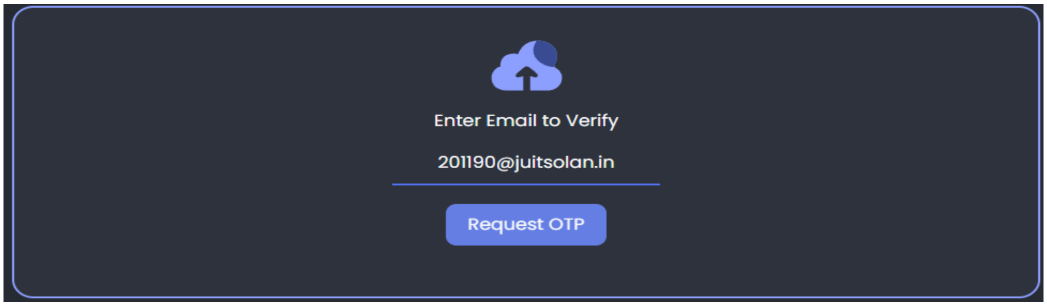
Fig. 3.2: User Interface

Backend Processing: This layer manages the system's essential functions, such as file encryption, compression, and cloud storage service integration. Using Express.js and Node.js together offers a scalable and reliable server-side architecture. Because of its event-driven, non-blocking I/O approach, Node.js is a good choice for concurrent file upload and processing activities since it guarantees high concurrency and responsiveness. Express.js makes it easier to create RESTful APIs, which makes it easier for frontend and backend components to communicate with each other.

Encryption Module: The module plays a crucial role in guaranteeing the security of data both during file transfer and storage. The system encrypts submitted files using strong encryption techniques like AES-CBC by using the crypto module in Node.js. By encrypting data in fixed-size blocks and chaining them together, AES-CBC (Advanced Encryption Standard - Cypher Block Chaining) protects confidentiality and integrity. By guaranteeing data isolation between users and limiting key reuse, this module's dynamic encryption key generation improves security.

Compression Module: Optimizing storage capacity and enhancing the effectiveness of data transport are greatly aided by file compression. The uploaded files are first compressed by the system using the ZLIB library in Node.js before being stored in the cloud. Zlib uses the popular compression method DEFLATE to minimize file size without compromising data integrity. The technology improves overall system performance by reducing storage costs and speeding up data transmission times by compressing files before storing them.

Email Verification: Email verification is necessary to confirm the legitimacy of user-provided email addresses and to authenticate users. To send users one-time passwords (OTPs) via verification emails, the system makes use of the Nodemailer module in Node.js. Nodemailer offers a stable and dependable API for communicating with SMTP servers, which streamlines the email sending process. The solution improves security by lowering the possibility of unauthorised access and confirming user identities through the integration of email verification.

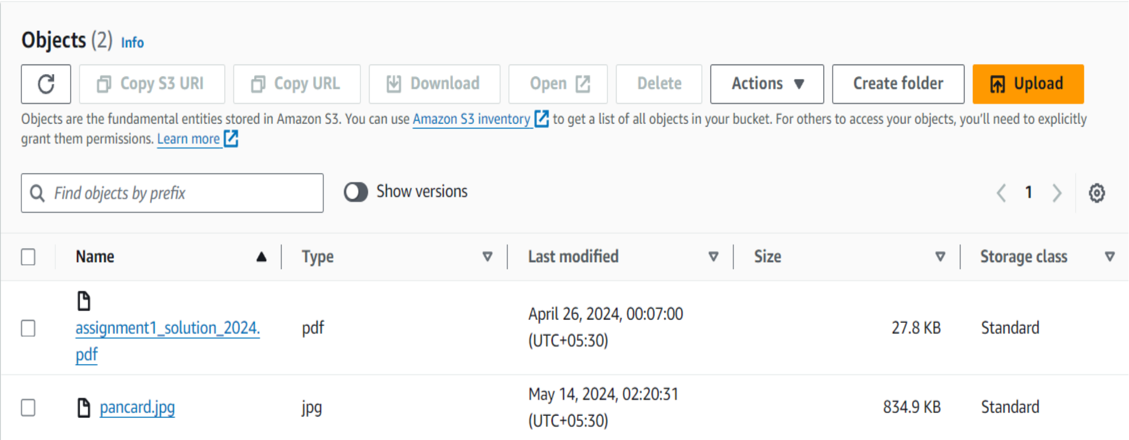


Enter Email to Verify

201190@juitsolan.in

Fig. 3.3 : Email Verification

Cloud Integration: Scalable and dependable data storage solutions require integration with cloud storage providers. To communicate with Amazon S3, a highly scalable object storage service provided by Amazon Web Services (AWS), the system makes use of the AWS SDK for JavaScript. Data accessibility and resilience are ensured by Amazon S3, which offers backup file storage that is safe, dependable, and highly accessible.



Objects (2) Info

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

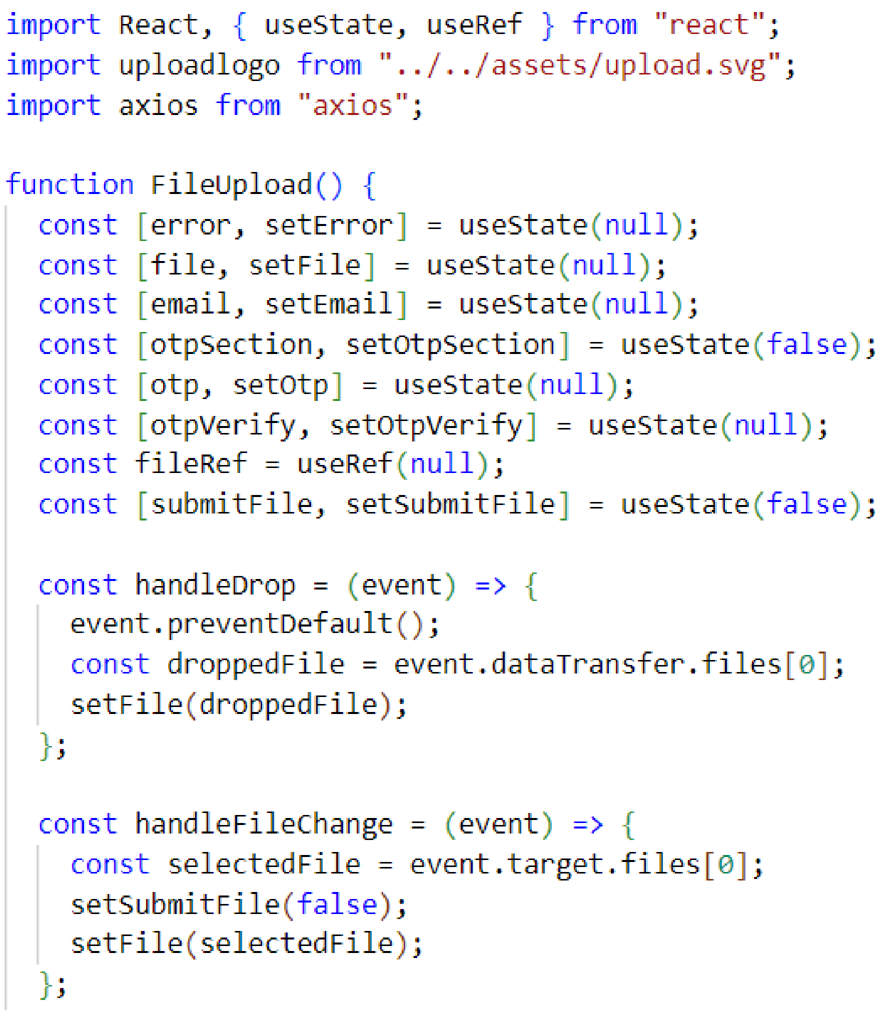
□	Name ▲	Type	Last modified	Size	Storage class
□	assignment1_solution_2024.pdf	pdf	April 26, 2024, 00:07:00 (UTC+05:30)	27.8 KB	Standard
□	pancard.jpg	jpg	May 14, 2024, 02:20:31 (UTC+05:30)	834.9 KB	Standard

Fig. 3.4 : AWS S3 Bucket

3.2.2 TOOLS, TECHNOLOGIES, AND LANGUAGES

The selected tools, technologies, and languages contribute to the robustness and efficiency of the system:

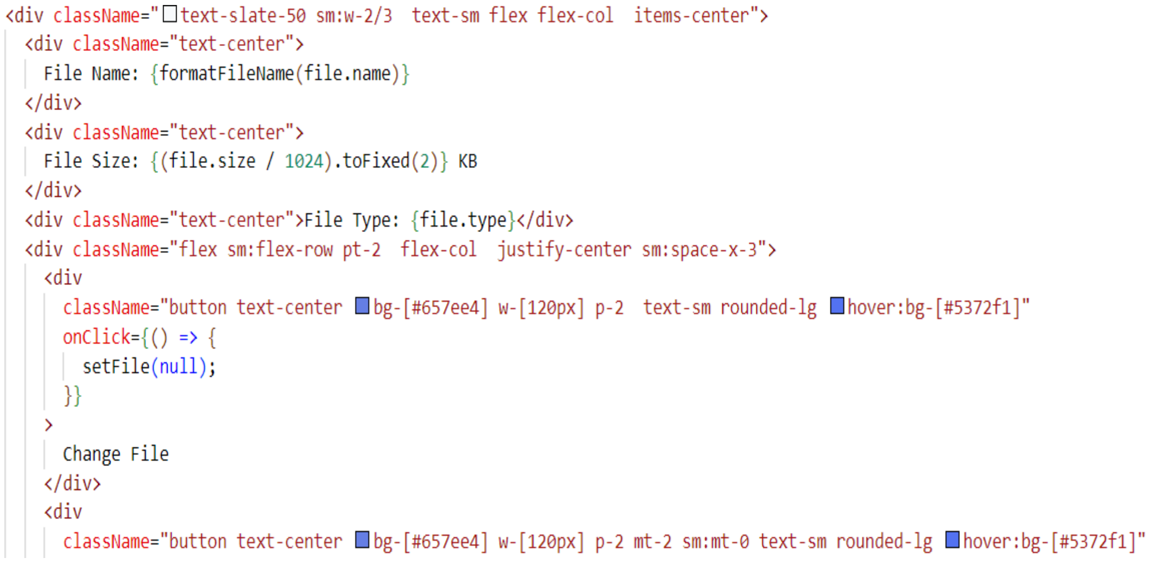
a. **React:** Facebook created React, a JavaScript library for creating user interfaces. Complex user interface management is made simpler by the ability for developers to construct reusable UI components. React's virtual Document Object Model (DOM) makes it possible to render UI components more effectively, which boosts responsiveness and speed. React's component-based design also encourages code maintainability and reusability, which makes it easier to create scalable and modular frontend apps.



```
import React, { useState, useRef } from "react";
import uploadlogo from "../../assets/upload.svg";
import axios from "axios";

function FileUpload() {
  const [error, setError] = useState(null);
  const [file, setFile] = useState(null);
  const [email, setEmail] = useState(null);
  const [otpSection, setOtpSection] = useState(false);
  const [otp, setOtp] = useState(null);
  const [otpVerify, setOtpVerify] = useState(null);
  const fileRef = useRef(null);
  const [submitFile, setSubmitFile] = useState(false);

  const handleDrop = (event) => {
    event.preventDefault();
    const droppedFile = event.dataTransfer.files[0];
    setFile(droppedFile);
  };

  const handleFileChange = (event) => {
    const selectedFile = event.target.files[0];
    setSubmitFile(false);
    setFile(selectedFile);
  };
```

Fig. 3.5 : React.js Code

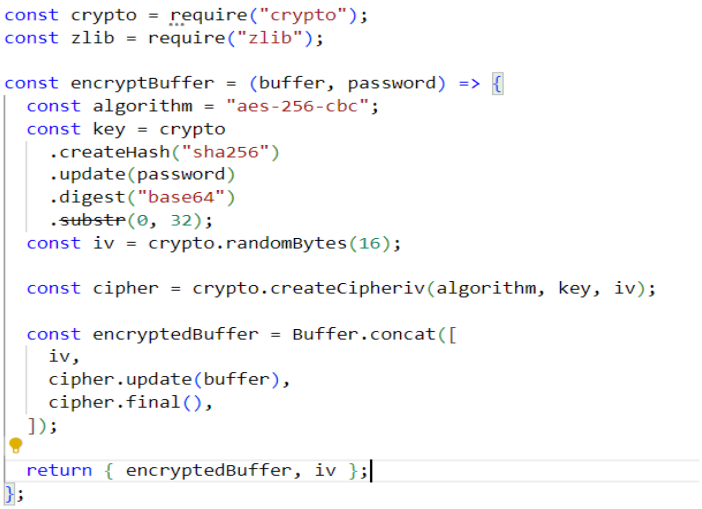
b. **Tailwind CSS:** Tailwind CSS is an HTML element styling system that prioritises utility above style. It offers a collection of predefined utility classes. Tailwind CSS does not impose any predetermined styles or components, in contrast to standard CSS frameworks, so developers may create bespoke designs without having to write unique CSS. Developers may quickly prototype and style UI components by employing utility classes directly in HTML markup. This leads to a more effective development process and consistent design patterns.



```
<div className="□text-slate-50 sm:w-2/3  text-sm flex flex-col  items-center">
  <div className="text-center">
    File Name: {formatFileName(file.name)}
  </div>
  <div className="text-center">
    File Size: {(file.size / 1024).toFixed(2)} KB
  </div>
  <div className="text-center">File Type: {file.type}</div>
  <div className="flex sm:flex-row pt-2  flex-col  justify-center sm:space-x-3">
    <div
      className="button text-center ■bg-[#657ee4] w-[120px] p-2  text-sm rounded-lg ■hover:bg-[#5372f1]"
      onClick={() => {
        setFile(null);
      }}
    >
      Change File
    </div>
    <div
      className="button text-center ■bg-[#657ee4] w-[120px] p-2 mt-2 sm:mt-0 text-sm rounded-lg ■hover:bg-[#5372f1]"
```

Fig. 3.6 : Tailwind CSS

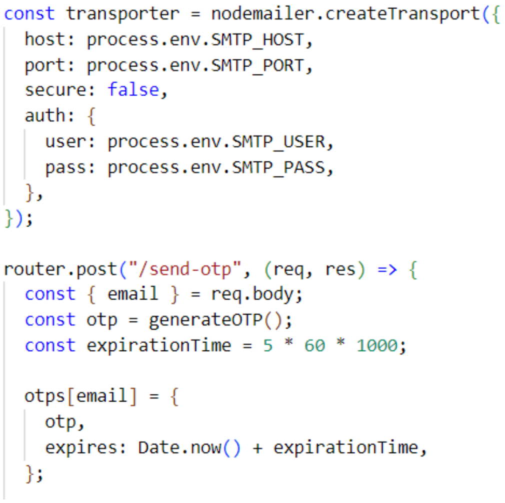
c. **Crypto Module:** Node.js's crypto module offers hashing, random number creation, encryption, and decryption among other cryptographic functions. The crypto module is used in the Secure Cloud Backup and Restore System to encrypt uploaded files prior to cloud storage. To guarantee data confidentiality and integrity during transmission and storage, the AES-CBC encryption technique is used. Sophisticated encryption algorithms and safe key management procedures are used by the system to shield private user information from hackers and illegal access.



```
const crypto = require("crypto");
const zlib = require("zlib");

const encryptBuffer = (buffer, password) => {
  const algorithm = "aes-256-cbc";
  const key = crypto
    .createHash("sha256")
    .update(password)
    .digest("base64")
    .~~substr~~(0, 32);
  const iv = crypto.randomBytes(16);

  const cipher = crypto.createCipheriv(algorithm, key, iv);

  const encryptedBuffer = Buffer.concat([
    iv,
    cipher.update(buffer),
    cipher.final(),
  ]);

  return { encryptedBuffer, iv };
};
```

Fig. 3.6 : Crypto Module

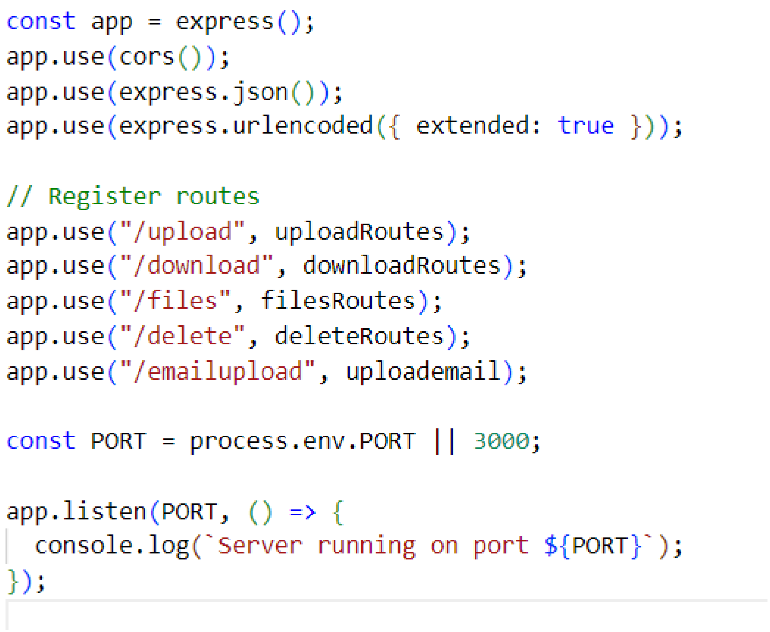
d. **Nodemailer:** Node.js's Nodemailer module lets programmers send emails from their apps. It offers a variety of delivery mechanisms, such as SMTP, Sendmail, and Amazon SES, and handles emails in both plaintext and HTML formats. Nodemailer is used in the Secure Cloud Backup and Restore System to send users verification emails during the email verification procedure. The solution improves user experience by delivering timely notifications and changes and guarantees dependable email delivery through the integration of Nodemailer.



```
const transporter = nodemailer.createTransport({
  host: process.env.SMTP_HOST,
  port: process.env.SMTP_PORT,
  secure: false,
  auth: {
    user: process.env.SMTP_USER,
    pass: process.env.SMTP_PASS,
  },
});

router.post("/send-otp", (req, res) => {
  const { email } = req.body;
  const otp = generateOTP();
  const expirationTime = 5 * 60 * 1000;

  otps[email] = {
    otp,
    expires: Date.now() + expirationTime,
  };
```

Fig. 3.7 : Nodemail

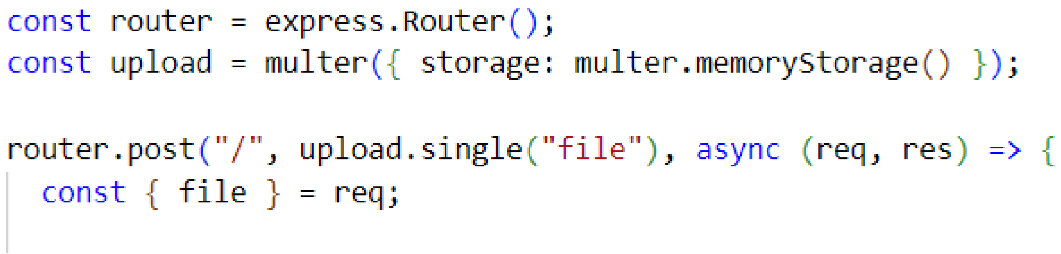
e. **Express:** Express.js is a Node.js web framework designed to be simple and lightweight, making it easier to create online apps and APIs. It offers a powerful feature set for managing middleware, routing, templating, and HTTP requests. Express.js is used in the Secure Cloud Backup and Restore System to create the backend server and define API routes. Because of its adaptable and lightweight design, frontend and backend components may communicate with each other seamlessly, allowing developers to construct server-side applications that are scalable.



```
const app = express();
app.use(cors());
app.use(express.json());
app.use(express.urlencoded({ extended: true }));

// Register routes
app.use("/upload", uploadRoutes);
app.use("/download", downloadRoutes);
app.use("/files", filesRoutes);
app.use("/delete", deleteRoutes);
app.use("/emailupload", uploademail);

const PORT = process.env.PORT || 3000;

app.listen(PORT, () => {
  console.log(`Server running on port ${PORT}`);
});
```

Fig. 3.8 : Express.js

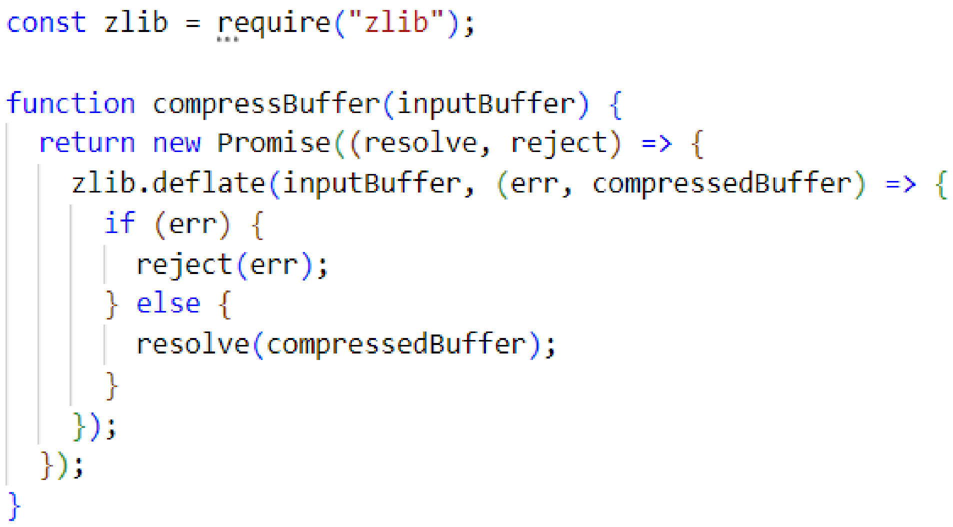
f. **Multer:** A popular middleware for Node.js applications, Multer handles multipart/form-data and is used to process file uploads. It offers features like file size restrictions, file type checking, and file storage settings, and it interfaces with Express.js effortlessly. Multer is used in Secure Cloud Backup and Restore System to manage file uploads from frontend to backend server. The technology guarantees dependable file processing and storage by streamlining the file uploading procedure with Multer.



```
const router = express.Router();
const upload = multer({ storage: multer.memoryStorage() });

router.post("/", upload.single("file"), async (req, res) => {
  const { file } = req;
```

Fig. 3.9 : Multer

- **g. Zlib:** Zlib is a Node.js compression library that supports DEFLATE, Gzip, and Brotli, among other compression algorithms. With its help, developers may effectively compress and decompress data streams, resulting in smaller files and faster data transmission rates. Zlib is used in the Secure Cloud Backup and Restore System to compress uploaded data prior to cloud storage. The system reduces costs and improves performance by optimizing storage capacity and improving data transport efficiency through file compression.



```
const zlib = require("zlib");

function compressBuffer(inputBuffer) {
  return new Promise((resolve, reject) => {
    zlib.deflate(inputBuffer, (err, compressedBuffer) => {
      if (err) {
        reject(err);
      } else {
        resolve(compressedBuffer);
      }
    });
  });
}
```

Fig. 3.10 : Zlib Library

The creation and functionality of the Secure Cloud Backup and Restore System depend heavily on each of these instruments, programming languages, and technological advancements. Together, they guarantee the system's performance, security, and dependability and make it easier for system components to communicate effectively and integrate with cloud platforms.

3.2.3 ARCHITECTURE

The project's goals of guaranteeing data security, scalability, and user-friendliness are all met by the selected architecture:

1. **SECURITY:** To protect user data from unauthorised access and data breaches, the system uses industry-standard encryption techniques and authentication procedures.

2. **SCALING:** Smooth scaling is made possible via integration with AWS, allowing for the expansion of the user base and the storage of larger amounts of backup data.

3. **EFFICIENCY:** By optimising storage capacity and boosting data transmission efficiency, the combination of compression techniques with cloud-based storage solutions improves system performance and user experience.

The foundation for the next stages of implementation and testing is set by this architectural design. Throughout the development lifespan, ongoing assessments and modifications will be made to satisfy any evolving requirements or improvements. The system attempts to provide a safe, scalable, and effective solution for cloud-based data backup and recovery by using state-of-the-art technology and best practices in system architecture.

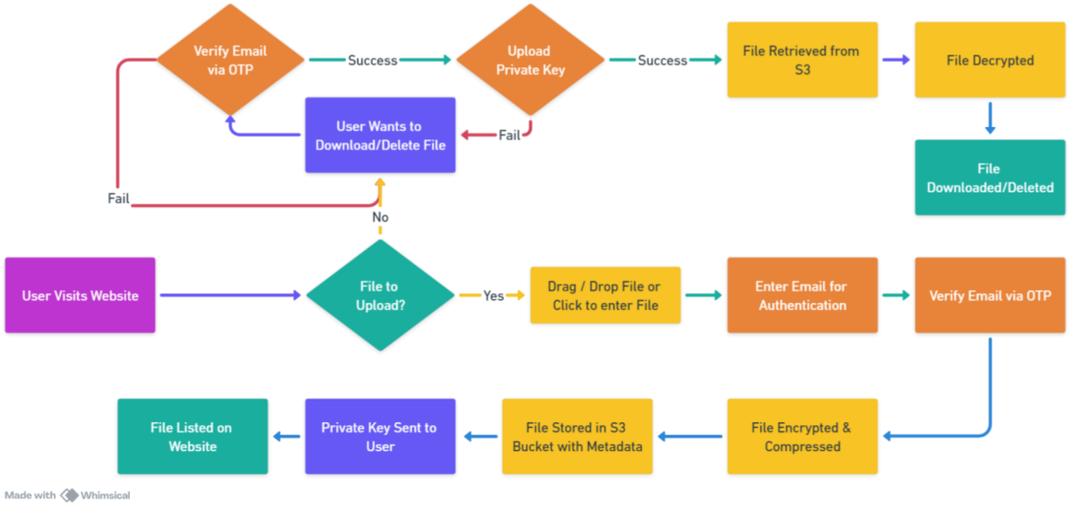
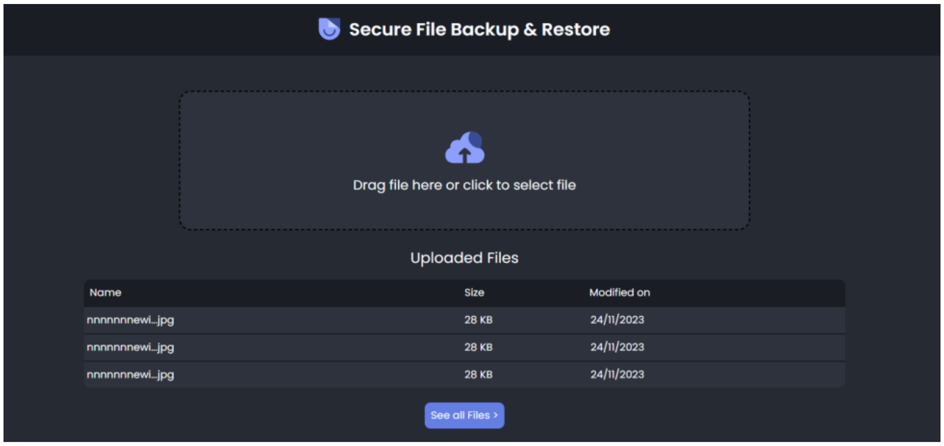


Fig 3.11 : Architecture of the project

3.3 IMPLEMENTATION

The implementation step involves transforming the design and architecture into a functional system. This section provides an overview of the essential implementation components, including the techniques applied for encryption (AES and RSA) and user login authentication.

1. **IMPROVED USER INTERFACE:** The frontend interface has been painstakingly designed to offer consumers a simple and efficient experience. The UI uses Tailwind CSS and React.js to provide a drag-and-drop page that makes file uploading easier. Uploading files is as simple as dragging them into the allotted space for users to provide a smooth and intuitive experience.


Secure File Backup & Restore
Drag file here or click to select file
Uploaded Files
See all Files ›

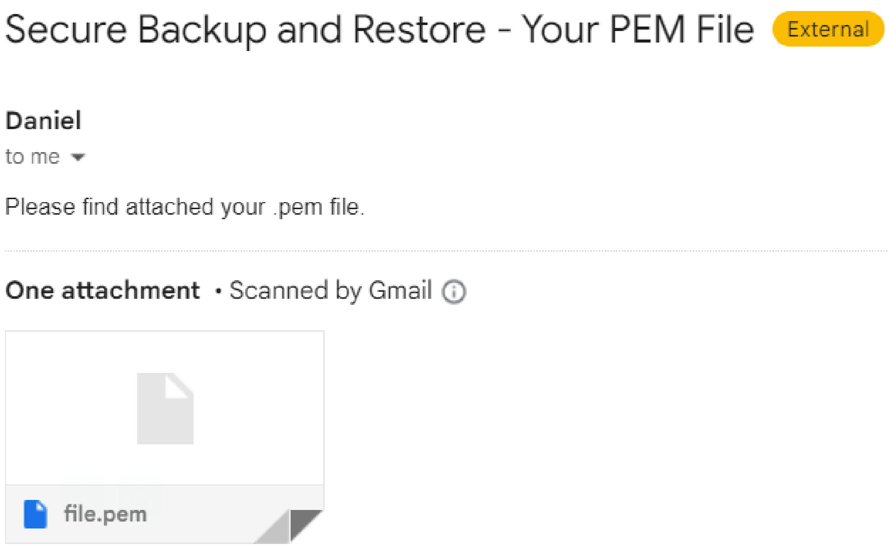
Name	Size	Modified on
nnnnnnnewi...jpg	28 KB	24/11/2023
nnnnnnnewi...jpg	28 KB	24/11/2023
nnnnnnnewi...jpg	28 KB	24/11/2023

Fig. 3.12 : Home UI

2. **MECHANISM FOR DATA COMPRESSION:** Node.js's zlib module is used to compress uploaded files. This compression method reduces the size of files before they are saved in the cloud, optimising storage space and enhancing data transport efficiency. The system optimises resource utilisation and guarantees peak speed while storing and retrieving data by employing file compression.

3. **ADVANCED ENCRYPTION OF DATA:** Modern encryption methods are used to protect the integrity and privacy of submitted files. Files are encrypted using the AES (Advanced Encryption Standard) technique by utilising the crypto module in Node.js.

Sensitive information is kept safe from unwanted access during transmission and storage thanks to this encryption procedure. Strong encryption measures are integrated into the system to ensure that it meets strict security requirements and reduces the possibility of data breaches.



Secure Backup and Restore - Your PEM File

Daniel

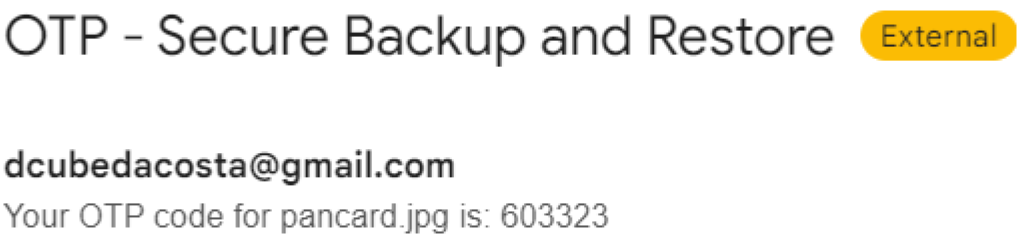
Please find attached your .pem file.

One attachment • Scanned by Gmail

file.pem

Fig. 3.13 : PEM File via mail

4. **EMAIL-BASED OTP USER AUTHENTICATION:** A multi-factor authentication strategy is used to strengthen user authentication, where users get One-Time Passwords (OTP) via email for validation. By using the Nodemailer module, the system creates and sends one-time passwords (OTPs) to users' email addresses, confirming their identity when they log in. Through an extra layer of verification and a decreased chance of unwanted access and security concerns, this authentication technique improves system security.



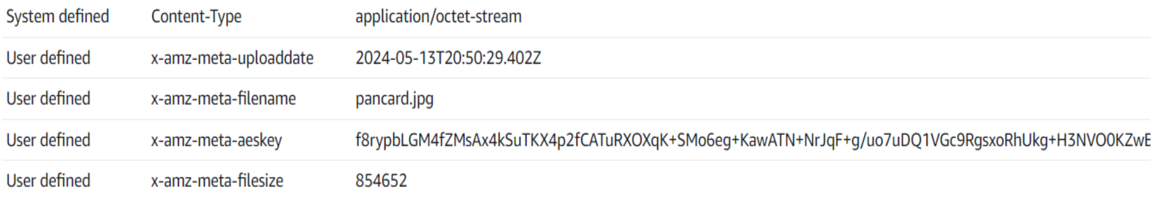
OTP - Secure Backup and Restore

dcubedacosta@gmail.com

Your OTP code for pancard.jpg is: 603323

Fig. 3.14 : OTP based user authentication

5. **SMOOTH INTEGRATION WITH AMAZON S3:** Amazon S3 (Simple Storage Service), a reliable and scalable cloud storage option, is smoothly integrated with the system. Files are safely kept in assigned S3 buckets using the AWS SDK, guaranteeing excellent availability, robustness, and accessibility. The system's scalability and dependability are improved by integration with AWS S3, which also allows for easy file uploading and retrieval while upholding strict data security guidelines.



System defined	Content-Type	application/octet-stream
User defined	x-amz-meta-uploaddate	2024-05-13T20:50:29.402Z
User defined	x-amz-meta-filename	pancard.jpg
User defined	x-amz-meta-aeskey	f8rypbLGM4fZMsAx4kSuTKX4p2fCATuRXOXqK+SMo6eg+KawATN+NrJqF+g/uo7uDQ1VGc9RgsxoRhUkg+H3NVO0KZwE
User defined	x-amz-meta-filesize	854652

Fig. 3.15 : AWS S3 Integration

6. **EASY-TO-USE FILE ORGANISATION:** The file management system has its own sites where users may see a list of files that have been uploaded and start downloads as needed. These intuitive user interfaces make uploaded files easily accessible, improving usability and enabling effective file management. Customers are guaranteed a smooth and simple user experience by being able to effortlessly browse through the files they have uploaded, examine file metadata, and start download operations.

The thorough installation of these parts and features highlights the system's dedication to providing a safe, effective, and intuitive platform for file management and cloud backup. Users can trust the Secure Cloud Backup and Restore System with their data since every part of its implementation has been painstakingly engineered to meet the highest standards of security, performance, and user satisfaction.

3.4 KEY CHALLENGES

The development of the Secure Cloud Backup and Recovery System was not without its share of problems. Overcoming these issues became vital to assuring the device's resilience, safety, and user-friendliness. This section covers the number one problem experienced at some point of the development technique and the techniques applied to cope with them.

1. **SELECTING THE BEST COMPRESSION ALGORITHM:** Choosing the best compression algorithm turned out to be a crucial task. To find the best answer, many algorithms like zlib, gzip, and Bzip2 have to be thoroughly evaluated. To guarantee effective file compression while preserving system performance, it was crucial to balance variables including compression ratios, processing speed, and resource use.

2. **FINDING A COMPROMISE BETWEEN ENCRYPTION AND COMPRESSION:** It was difficult to strike the ideal balance between encryption and compression. It required great thought to implement strong encryption techniques like AES while effectively compressing encrypted material. Extensive research and testing were needed to optimise the compression and encryption settings to reduce overhead while maintaining data security.

3. **FLOWING INTEGRATION WITH CLOUD ENVIRONMENTS:** Considerable complexity was added when integrating the system smoothly with cloud systems like AWS S3 and Google Cloud Storage. There were issues in maintaining security, scalability, and reliability criteria while guaranteeing seamless interaction with cloud storage providers. It required a thorough understanding and efficient implementation techniques to adjust to various APIs, authentication methods, and data transport protocols.

4. **SCALABILITY TO MEET INCREASING NEEDS:** One of the most important challenges was designing the system to scale well to meet growing data quantities and user expectations. Careful design and execution were required to implement scalable architecture, distributed computing approaches, and reliable load balancing mechanisms to manage growing workloads while maintaining performance and dependability. Meeting scalability objectives required ensuring smooth scalability to handle changing resource demands and usage patterns.

5. **CONSTANT PERFORMANCE OPTIMISATION:** There were constant problems in maximising system performance across different components and functions. Ongoing optimisation efforts were required to fine-tune data structures, algorithms, and system configurations to maximise efficiency and minimise resource use. It took a proactive strategy and continuous improvement to balance performance improvements with changing user needs and technology breakthroughs.

CHAPTER 4: TESTING

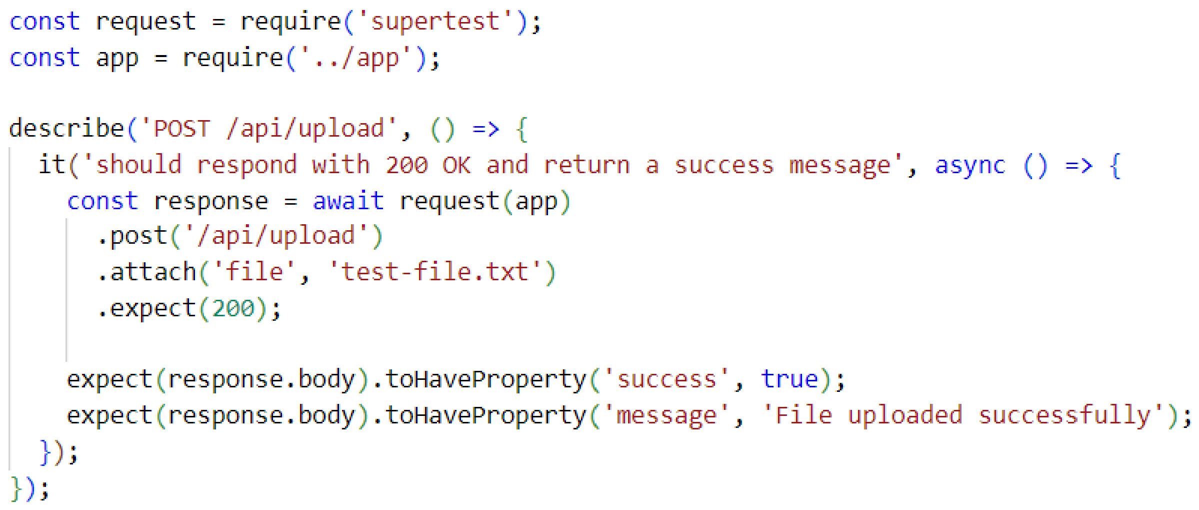
4.1 TESTING STRATEGY

To ensure the Secure Cloud Backup and Recovery System's dependability, security, and performance, a rigorous testing plan is necessary. This section outlines the tools employed for efficient testing across various dimensions as well as the testing strategy employed during the project.

4.1.1 LEVELS OF TESTING

UNIT TESTING:

- Unit tests will be conducted to validate the functionality of individual components, such as backend APIs, encryption algorithms, and authentication mechanisms.
- Test cases will cover edge cases, boundary conditions, and error handling scenarios to ensure robustness and reliability.

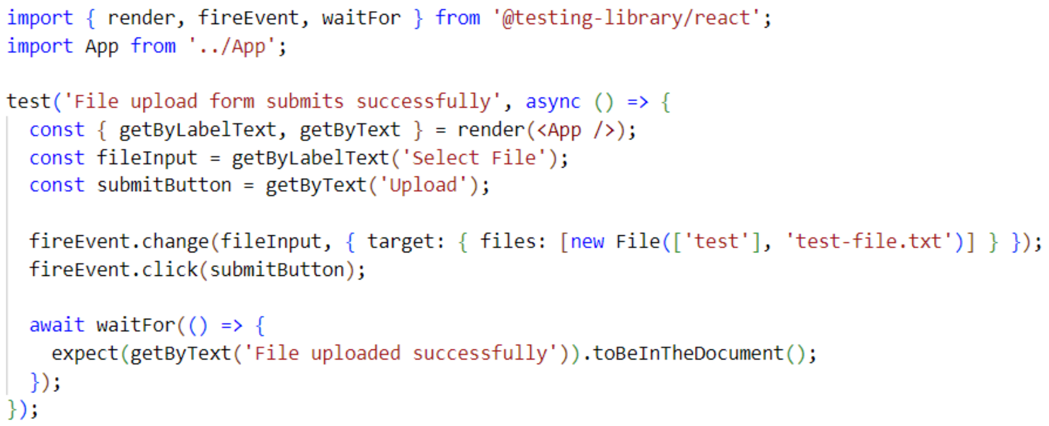


```
const request = require('supertest');
const app = require('../app');

describe('POST /api/upload', () => {
  it('should respond with 200 OK and return a success message', async () => {
    const response = await request(app)
      .post('/api/upload')
      .attach('file', 'test-file.txt')
      .expect(200);

    expect(response.body).toHaveProperty('success', true);
    expect(response.body).toHaveProperty('message', 'File uploaded successfully');
  });
});
```

Fig. 4.1 : Unit Testing

INTEGRATION TESTING:

- Integration tests will verify the interactions between different system components, including frontend and backend integration, API endpoints, and third-party services.
- Test suites will be designed to validate data flow, communication protocols, and error handling across interconnected modules.

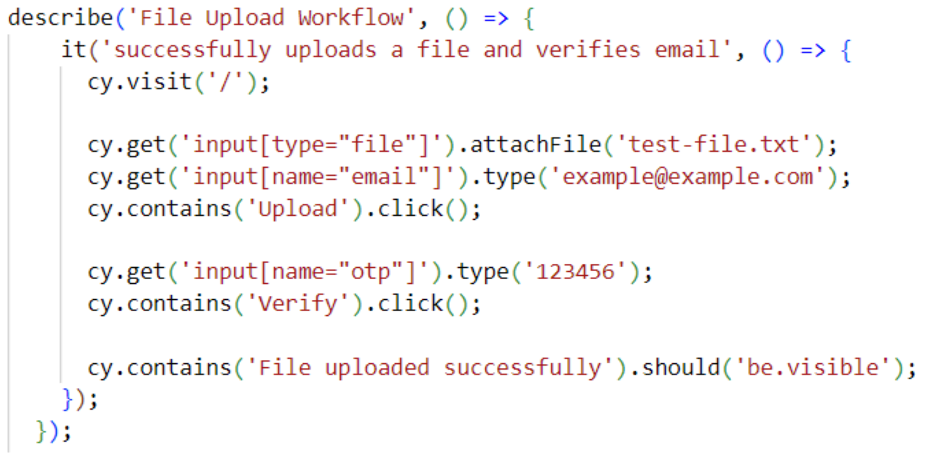


```
import { render, fireEvent, waitFor } from '@testing-library/react';
import App from '../App';

test('File upload form submits successfully', async () => {
  const { getByLabelText, getByText } = render(<App />);
  const fileInput = getByLabelText('Select File');
  const submitButton = getByText('Upload');

  fireEvent.change(fileInput, { target: { files: [new File(['test'], 'test-file.txt')] } });
  fireEvent.click(submitButton);

  await waitFor(() => {
    expect(getByText('File uploaded successfully')).toBeInTheDocument();
  });
});
```

Fig. 4.2 : Integration Testing

END TO END(E2E) TESTING:

- End-to-end tests will simulate real-world user scenarios to validate the entire system's functionality from user interaction to backend processing and cloud storage.
- E2E test cases will cover common user workflows, such as file upload, download, verification, and deletion, ensuring seamless operation across all system components.

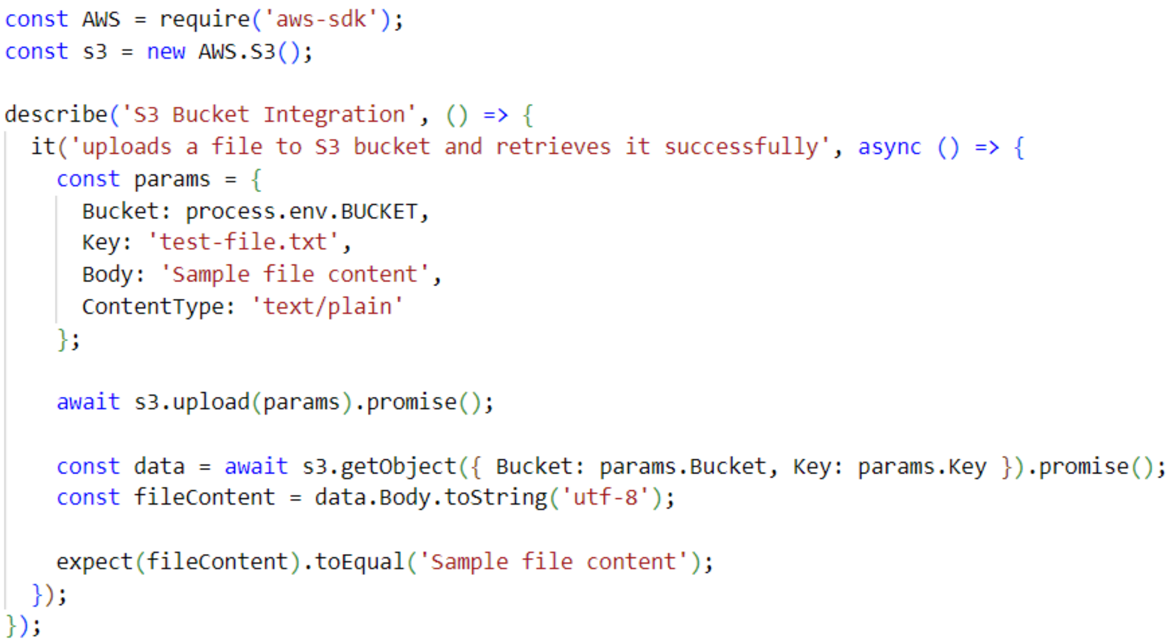


```
describe('File Upload Workflow', () => {
    it('successfully uploads a file and verifies email', () => {
      cy.visit('/');

      cy.get('input[type="file"]').attachFile('test-file.txt');
      cy.get('input[name="email"]').type('example@example.com');
      cy.contains('Upload').click();

      cy.get('input[name="otp"]').type('123456');
      cy.contains('Verify').click();

      cy.contains('File uploaded successfully').should('be.visible');
    });
  });
```

Fig. 4.3 : E2E Testing

S3 BUCKET TESTING:

- Test the integration with the S3 bucket to ensure that files are being securely stored and retrieved.
- Verify that files are compressed before being uploaded to the S3 bucket to minimise storage costs and optimise transfer speeds.
- Test the scalability of the system by uploading a large number of files to the S3 bucket and monitoring performance metrics.

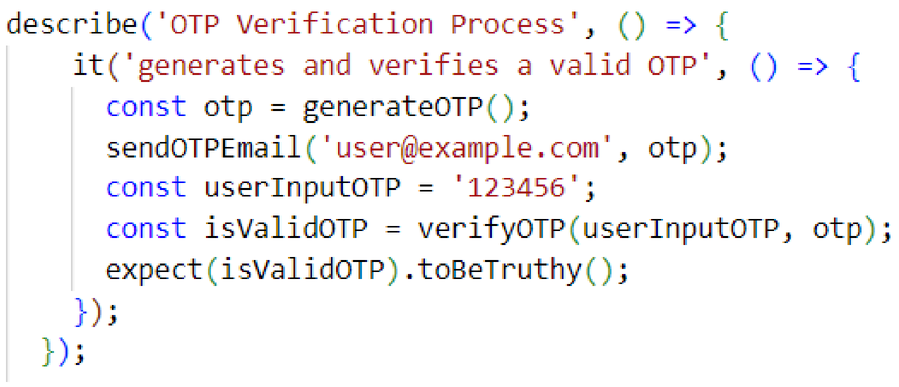


```
const AWS = require('aws-sdk');
const s3 = new AWS.S3();

describe('S3 Bucket Integration', () => {
  it('uploads a file to S3 bucket and retrieves it successfully', async () => {
    const params = {
      Bucket: process.env.BUCKET,
      Key: 'test-file.txt',
      Body: 'Sample file content',
      ContentType: 'text/plain'
    };

    await s3.upload(params).promise();

    const data = await s3.getObject({ Bucket: params.Bucket, Key: params.Key }).promise();
    const fileContent = data.Body.toString('utf-8');

    expect(fileContent).toEqual('Sample file content');
  });
});
```

Fig. 4.4 : AWS S3 Testing

OTP VERIFICATION TESTING:

- Test the OTP generation process to ensure that unique and time-limited OTPs are generated for each email verification request.
- Verify that the OTP is sent to the correct email address and that it can be successfully retrieved by the user.
- Test the OTP validation process to ensure that users can verify their email addresses securely before proceeding with file uploads.

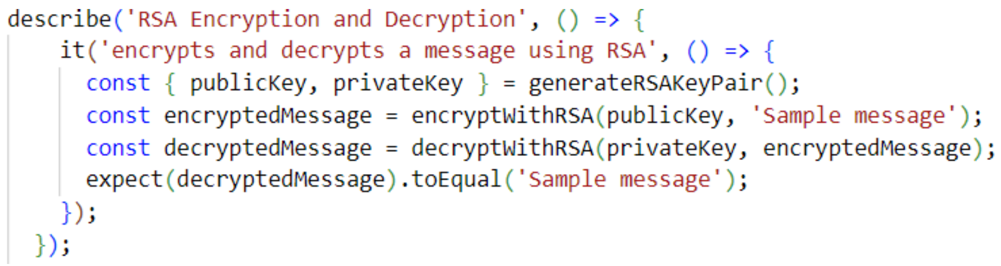


```
describe('OTP Verification Process', () => {
  it('generates and verifies a valid OTP', () => {
    const otp = generateOTP();
    sendOTPEmail('user@example.com', otp);
    const userInputOTP = '123456';
    const isValidOTP = verifyOTP(userInputOTP, otp);
    expect(isValidOTP).toBeTruthy();
  });
});
```

Fig. 4.5 : OTP Verification

RSA KEY GENERATION TESTING:

- Test the RSA key pair generation process to ensure that secure key pairs are generated with sufficient key lengths (e.g., 2048 bits or higher).
- Verify that the public key is stored securely alongside file metadata and that the private key is securely transmitted to users for decryption.
- Test the RSA encryption and decryption process to ensure that files can be encrypted with the public key and decrypted with the private key successfully.



```
describe('RSA Encryption and Decryption', () => {
  it('encrypts and decrypts a message using RSA', () => {
    const { publicKey, privateKey } = generateRSAKeyPair();
    const encryptedMessage = encryptWithRSA(publicKey, 'Sample message');
    const decryptedMessage = decryptWithRSA(privateKey, encryptedMessage);
    expect(decryptedMessage).toEqual('Sample message');
  });
});
```

Fig.4.6 : RSA Key Generation

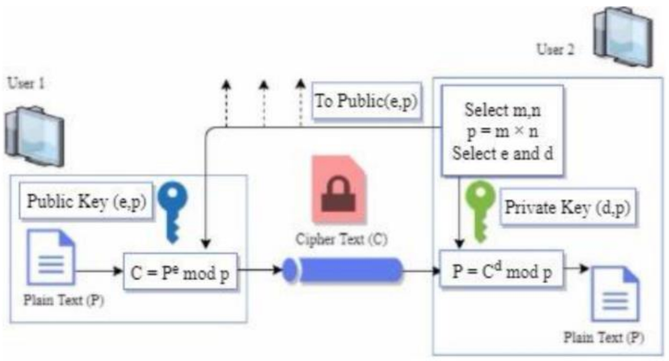
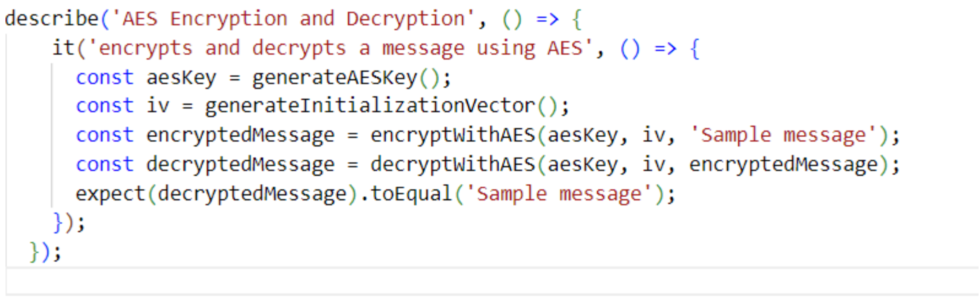


Fig. 4.7 : RSA Structure

AES ENCRYPTION TESTING:

- Test different AES encryption modes, such as ECB (Electronic Codebook), CBC (Cipher Block Chaining), and GCM (Galois/Counter Mode), to evaluate their performance and security characteristics.
- Verify that files encrypted using different AES encryption methods can be securely stored, retrieved, and decrypted without data loss or corruption.
- Test the impact of varying key sizes and initialization vectors (IVs) on the security and performance of AES encryption.



```
describe('AES Encryption and Decryption', () => {
    it('encrypts and decrypts a message using AES', () => {
      const aesKey = generateAESKey();
      const iv = generateInitializationVector();
      const encryptedMessage = encryptWithAES(aesKey, iv, 'Sample message');
      const decryptedMessage = decryptWithAES(aesKey, iv, encryptedMessage);
      expect(decryptedMessage).toEqual('Sample message');
    });
  });
```

Fig. 4.8 : AES encryption code

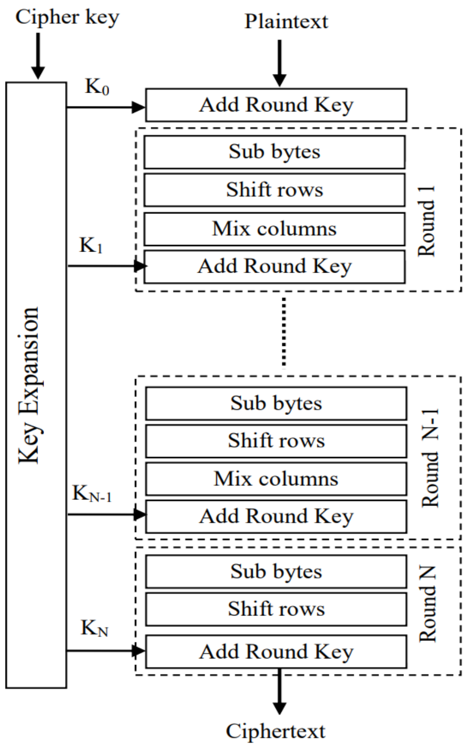


Fig. 4.9 : AES Structure

The testing phase confirmed the Secure Cloud Backup and Restore System's functionality, reliability, and security. With successful unit testing, integration testing, end-to-end testing, and cloud platform evaluations, the system demonstrated its capability to securely backup and restore files while maintaining optimal performance and scalability. These results provide confidence in the system's readiness for deployment and utilization in real-world scenarios, ensuring data protection and accessibility for users.

CHAPTER 5: RESULT AND EVALUATION

5.1 RESULTS

The comprehensive evaluation of the Secure Cloud Backup and Restore System provided valuable insights into its performance, reliability, and usability. Here, we delve deeper into the key findings and observations derived from the evaluation process.

5.1.1 PREFORMANCE ANALYSIS:

Performance evaluation focused on assessing the system's efficiency in file handling, compression, encryption, and data transfer processes.

1. **SEQUENTIAL COMPRESSION BEFORE ENCRYPTION:** Adopting a sequential approach of compression before encryption yielded notable performance improvements. By compressing files prior to encryption, the system achieved enhanced compression ratios, resulting in reduced storage requirements and improved data transfer speeds. This optimised sequence ensured efficient resource utilisation and minimised processing.

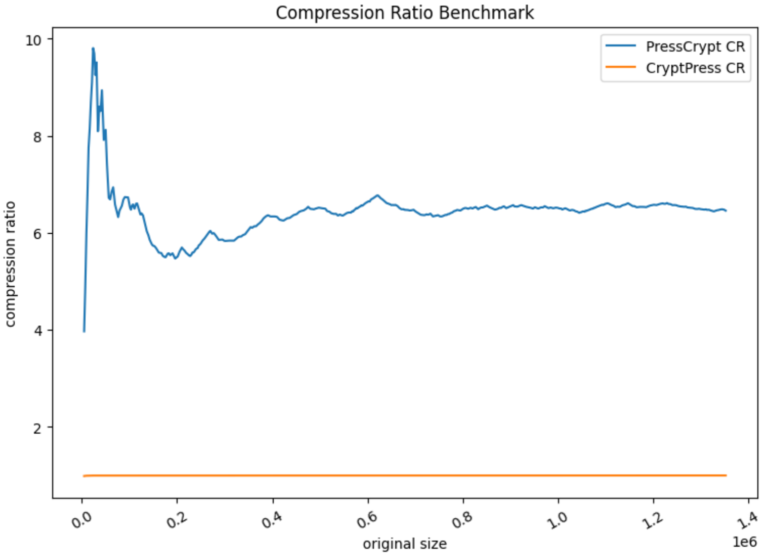


Fig 5.1 : Compression Ratio Benchmark

2. **AES-CBC ENCRYPTION PERFORMANCE:** The utilisation of AES-CBC encryption showcased commendable performance, effectively securing data while maintaining satisfactory processing speeds. The encryption process demonstrated robustness in safeguarding data integrity and confidentiality, enhancing the overall security posture of the system.

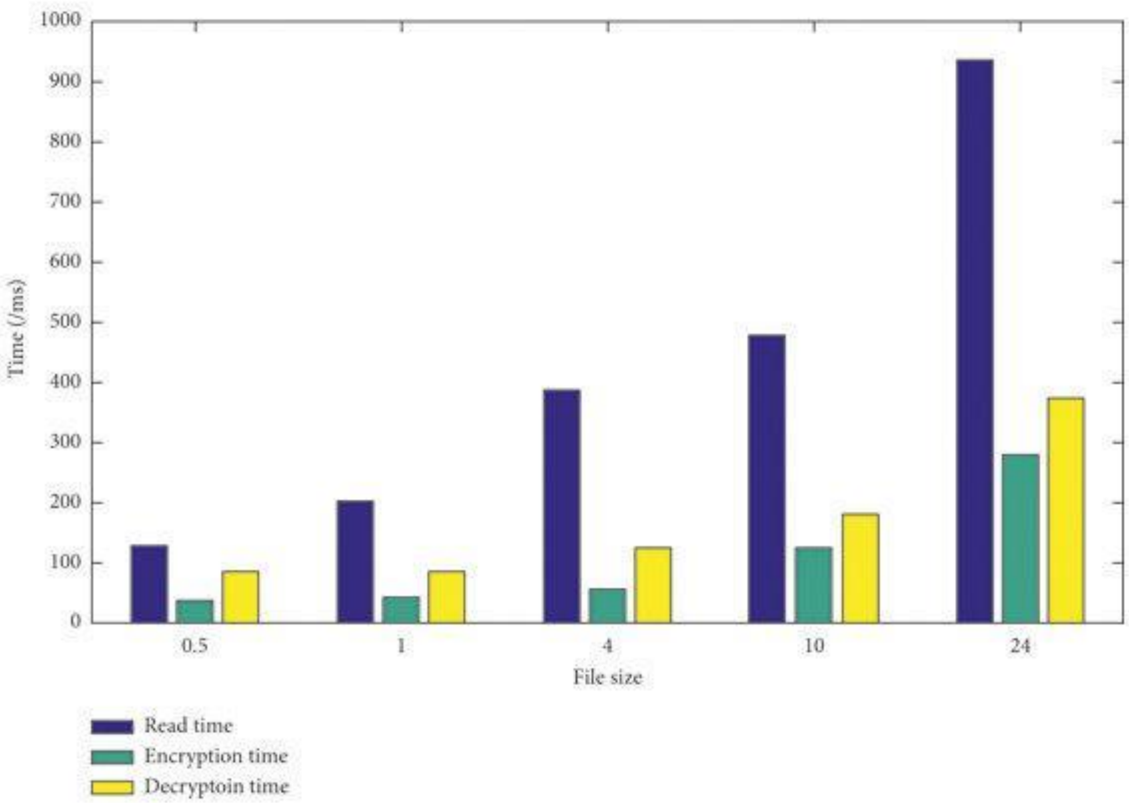


Fig 5.2 : AES-CBC Encryption Performance

5.1.2. RELIABILITY AND SECURITY ASSESSMENT

Ensuring the reliability and security of backed-up data was paramount to the system's functionality and trustworthiness.

1. **DATA INTEGRITY PRESERVATION:** The decision to compress files before encryption proved instrumental in preserving data integrity throughout the backup and recovery process. Files underwent compression without compromising data integrity,

ensuring that compressed data remained intact during encryption and subsequent decryption processes.

2. **AES-CBC ENCRYPTION SECURITY:** AES-CBC encryption emerged as a robust security measure, effectively protecting sensitive data against unauthorised access and manipulation. The adoption of strong encryption mechanisms bolstered data confidentiality, instilling confidence in the system's ability to safeguard critical information assets.

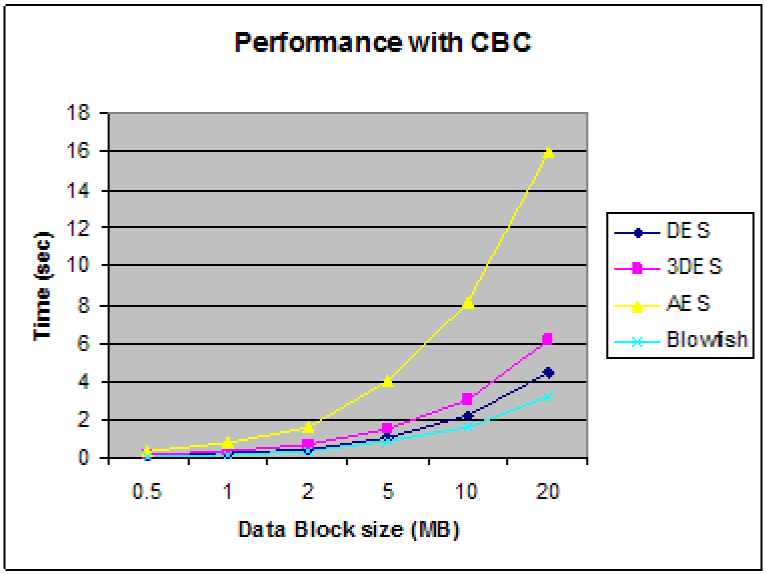


Fig 5.3 : Performance with CBC

5.1.3. USER EXPERIENCE AND USABILITY EVALUATION

User-centric aspects such as interface intuitiveness, authentication mechanisms, and overall user satisfaction were thoroughly assessed.

1. **SEAMLESS FILE MANAGEMENT:** Users lauded the system's intuitive file management capabilities, including the drag-and-drop functionality for file upload and the streamlined process for file retrieval. The user-friendly interface contributed to a seamless experience, facilitating efficient file management operations.

2. **EFFICIENT AUTHENTICATION MECHANISMS:** User authentication via OTP sent to email addresses received positive feedback for its effectiveness and reliability. The implementation of OTP verification enhances security while ensuring user convenience, thereby fostering trust and confidence in the system.

5.1.4. SCALABILITY AND RESOURCE OPTIMIZATION

Scalability and resource management were critical considerations to ensure the system's adaptability to varying workloads and resource constraints.

Scalable Architecture: Integration with cloud platforms such as AWS S3 provided the system with inherent scalability, enabling it to accommodate growing data volumes and user demands seamlessly. The scalable architecture ensured that the system could efficiently handle increased workloads without compromising performance or reliability.

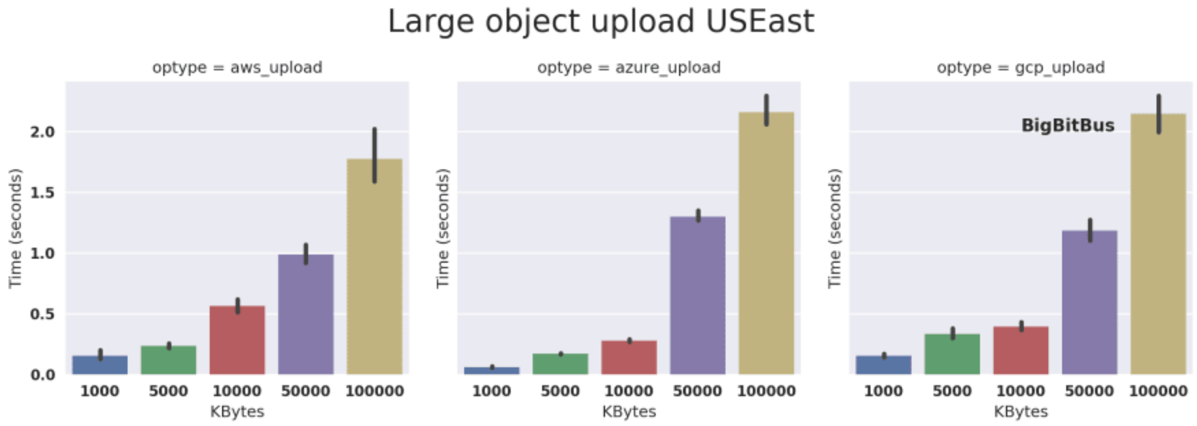


Fig. 5.4 : Upload speed different size

Resource Optimization: The system demonstrated efficient resource management, effectively utilising memory and processing resources to optimise performance. Resource utilisation was streamlined to minimise overhead and ensure optimal operation under diverse workload conditions, enhancing the system's efficiency and responsiveness.

The results of the evaluation underscored the effectiveness and reliability of the Secure Cloud Backup and Restore System. Through sequential compression before encryption, the system achieved enhanced data compression without compromising security, ensuring data integrity and confidentiality throughout the backup and recovery processes. The user-centric design, robust security measures, and scalable architecture position the system as a trusted solution for secure data management in cloud environments. Moving forward, continued optimization and refinement will be essential to sustain the system's performance, security, and user satisfaction in dynamic operational environments.

CHAPTER 6:
CONCLUSIONS AND FUTURE SCOPE

6.1 CONCLUSION

The creation of the Secure Cloud Backup and Recovery System has been an extended journey, weaving through research, design, implementation, and testing. This conclusion captures the significant results, admits limits, and delineates the substantial contributions made in the realms of data security and cloud-based systems.

6.1.1 KEY FINDINGS:

1. **SECURITY AND ENCRYPTION:** The use of solid encryption methods, such as AES and RSA, offers a stalwart basis for maintaining the maximum data secrecy in the delicate domains of backup and recovery procedures.
2. **AUTHENTICATION AND AUTHORIZATION:** The adoption of secure authentication techniques and thorough permission rules fortifies the entire system's resilience, functioning as a powerful bulwark against any invasion of illegal access.
3. **CLOUD PLATFORM INTEGRATION:** Meticulous selection and fine-tuning of cloud platforms, particularly AWS, impart a scalable and stable infrastructure that seamlessly hosts the complexity of the system's design.
4. **TESTING AND QUALITY ASSURANCE:** The implementation of a holistic testing method, encompassing several tiers and dimensions, shows to be crucial in the discernment and subsequent mitigation of possible vulnerabilities and performance bottlenecks.
5. **RSA VULNERABILITY ANALYSIS:** Delving into possible attacks against RSA casts a discriminating light on the crucial need of developing and adhering to secure key management methods, therefore bolstering the cryptographic backbone of the system.

6. **KEY GENERATION SPEED:** The intentional and cautious speed of key generation in RSA presents itself as a possible restriction, particularly in circumstances involving the development of a significant volume of cryptographic keys.
7. **DECRYPTION SPEED:** The meditative tempo of the decryption process in RSA bears study, as its relative slowness may pose an influence on the system's overall efficacy, particularly when dealing with huge communications.
8. **AWS-SPECIFIC CHALLENGES:** The complex problems interwoven with RSA key generation and the speed of encryption/decryption inside the AWS environment need painstaking attention for optimal orchestration of the system's performance.
9. **SECURITY EDUCATION:** The critical topic of user education on security measures emerges as a focus point. The victory of the system pivots on users comprehending and resolutely adhering to security procedures in their interactions with the system.

6.1.2 CONTRIBUTIONS TO THE FIELD:

1. **ENCRYPTED CLOUD BACKUP AND RECOVERY SYSTEM:** The consummation of the project bequeaths to the realm a complicated and sophisticated technology, standing as a paragon in handling crucial problems of data security, privacy, and the flawless orchestration of recovery operations inside cloud settings.
2. **MULTI-DIMENSIONAL APPROACH:** The combination of encryption, cloud technology, software development, and user experience design highlight a multi-dimensional approach, capturing the essence of a comprehensive and resilient solution.
3. **RSA VULNERABILITY ANALYSIS:** The sophisticated examination of possible assaults on RSA stands as a remarkable addition, complementing the larger knowledge of the algorithm's resilience and vulnerability in the modern tapestry of data security settings.
4. **AWS INTEGRATION INSIGHTS:** Illuminating the problems and concerns essential to AWS integration, the project provides useful insights, acting as a beacon for future attempts in the landscape of cloud-based systems.

In summary, the Secure Cloud Backup and Recovery System serves as a tribute to the precise interaction of security, encryption, and cloud technology. While certain limits linger, the project's contributions to the field resound as a harmonic song, setting the way for future advancements in the domains of safe, scalable, and user-centric cloud-based systems.

6.2 FUTURE SCOPE

The future of the Secure Cloud Backup and Recovery System is going to be focusing on the areas of deduplication, user login access, and the integration of compression queues such as RabbitMQ and Docker images for server accessibility, all of which will lead to great advancements in data management, accessibility and disaster recovery.

- **ADVANCED DEDUPLICATION TECHNIQUES:** Later versions of the system could improve the deduplication methods and thus the storage efficiency will be more. The application of the new algorithms, for example, the content-aware or the inline deduplication, could reduce the duplicate data storage by identifying and removing the redundant files or blocks. Thus, it would be the saving of a lot of storage space and bandwidth utilisation, especially in large-scale backup operations.
- **ENHANCED USER LOGIN ACCESS:** The next stages of the system may be directed towards the enhancement of user authentication and access control systems. The introduction of strong user login systems, in particular, the multi-factor authentication and the role-based access control, would be a great way to prevent unauthorised access to the sensitive backup data by authorised users only. This will make the system more resistant to hack attempts and data breaches.
- **INTEGRATION OF COMPRESSION QUEUES:** Compression queues like RabbitMQ can be added into the system architecture to make data processing and transmission tasks easier and more efficient. With the help of message queuing technology, the system is able to handle and prioritise the backup and recovery tasks in an efficient way, thus, providing the timely and reliable data transfers. Such a move would improve the system scalability and responsiveness, particularly during the times of high workload or when dealing with large amounts of data.

- **DEPLOYMENT OF DOCKER IMAGE OF SERVER FOR EASIER ACCESS:** Containing the server system's components through the use of containerization technology, such as Docker, paves the way for the easier deployment and management of the system. The use of Docker images for the server application will result in the seamless deployment of the application across different computing environments, thus guaranteeing the consistency and the portability of the system. This method makes the system more available for both administrators and end-users, which in turn, allows them to set up and manage the backup and recovery infrastructure in a more efficient way.

To sum up, the Secure Cloud Backup and Recovery System's path in the future is full of opportunities. Through the development of these pathways, the system is ready not only to deal with the emerging needs of the users but also to take the first step in the adoption of new technologies in the constantly changing field of data management and security.

REFERENCES

[1] G. Ramesh , J. Logeshwaran, and V. Aravindarajan, “A Secured Database Monitoring Method to Improve Data Backup and Recovery Operations in Cloud Computing,” 2023, BOHR International Journal of Computer Science, Vol. , No. 1, pp. 1–7, DOI: 10.54646/bijcs.019

[2] R. Adee, H. A Mouratidis, “Dynamic Four-Step Data Security Model for Data in Cloud Computing Based on Cryptography and Steganography,” Sensors 2022, 22, 1109, DOI: 10.3390/s22031109

[3] D. Chang, L. Li, Y. Chang, and Z. Qiao, "Cloud Computing Storage Backup and Recovery Strategy Based on Secure IoT and Spark," Mobile Information Systems, vol. 2021, Article ID 9505249, pp. 1-13, 2021. DOI: 10.1155/2021/9505249

[4] R. Maher and O. A. Nasr, "DropStore: A Secure Backup System Using Multi-Cloud and Fog Computing," in IEEE Access, vol. 9, pp. 71318-71327, 2021, doi: 10.1109/ACCESS.2021.3078887.

[5] A. A. Mughal, “Cybersecurity Architecture for the Cloud: Protecting Network in a Virtual Environment,” IJIAC, vol. 4, no. 1, pp. 35–48, Mar. 2021.

[6] S. Modi, Y. Dakwala, and V. Panchal, "Cloud Backup & Recovery Techniques of Cloud Computing and a Comparison between AWS and Azure Cloud," International Research Journal of Engineering and Technology (IRJET), vol. 07, no. 07, pp. 1897, July 2020, e-ISSN: 2395-0056, p-ISSN: 2395-0072.

[7] A. A. Tamimi, R. Dawood and L. Sadaqa, "Disaster Recovery Techniques in Cloud Computing," 2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT), Amman, Jordan, 2019, pp. 845-850, doi: 10.1109/JEEIT.2019.8717450.

[8] W. Sun, N. Zhang, W. Lou and Y. T. Hou, "Tapping the Potential: Secure Chunk-based Deduplication of Encrypted Data for Cloud Backup," 2018 IEEE Conference on Communications and Network Security (CNS), Beijing, China, 2018, pp. 1-9, doi: 10.1109/CNS.2018.8433173.

[9] V. Chang, "Towards a Big Data system disaster recovery in a Private Cloud," Ad Hoc Networks, vol. 35, pp. 65-82, Jul. 2015. DOI: 10.1016/j.adhoc.2015.07.012.

[10] S. Suguna and A. Suhasini, "Overview of data backup and disaster recovery in cloud," International Conference on Information Communication and Embedded Systems (ICICES 2014), Chennai, India, 2014, pp. 1-7, doi: 10.1109/ICICES.2014.7033804.

[11] M. Carroll, A. van der Merwe and P. Kotzé, "Secure cloud computing: Benefits, risks and controls," 2011 Information Security for South Africa, Johannesburg, South Africa, pp. 1-9, doi: 10.1109/ISSA.2011.6027519.

[12] A. Rahumed, H. C. H. Chen, Y. Tang, P. P. C. Lee and J. C. S. Lui, "A Secure Cloud Backup System with Assured Deletion and Version Control," 2011 40th International Conference on Parallel Processing Workshops, Taipei, Taiwan, 2011, pp. 160-167, doi: 10.1109/ICPPW.2011.17.

[13] S. Swagatika and N. Panda, “Cloud-based backup and data recovery,” Journal of Information and Optimization Sciences, vol. 43, no. 5, pp. 923–932, Jul. 2022, doi: 10.1080/02522667.2022.2091097.

[14] Z. Q. Wu and H. Li, “Analysis of data backup and recovery system,” Applied Mechanics and Materials, vol. 631–632, pp. 1207–1210, Sep. 2014, doi: 10.4028/www.scientific.net/amm.631-632.1207.

[15] P. S. Challagidad, A. S. Dalawai, and M. N. Birje, “Efficient and reliable data recovery technique in cloud computing,” The Internet of Things, vol. 5, no. 5, p. 13, Aug. 2017, doi: 10.11648/j.iotcc.s.2017050501.13.

[16] M. M. Alshammari, A. A. Alwan, A. Nordin, and I. F. T. Alshaikhli, “Disaster recovery in single-cloud and multi-cloud environments: Issues and challenges,” 4th IEEE International Conference on Engineering Technologies and Applied Sciences (ICETAS), Nov. 2017, doi: 10.1109/icetas.2017.8277868.

[17] L. Tawalbeh, N. S. Darwazeh, R. S. Al-Qassas, and F. Aldosari, “A Secure Cloud Computing Model based on Data Classification,” Procedia Computer Science, vol. 52, pp. 1153–1158, Jan. 2015, doi: 10.1016/j.procs.2015.05.150.

[18] R. K. Wadhwa and K. Sharma, "The commercial hard disk backup system for quick recovery operating system in cloud storage system," 2022 International Interdisciplinary Humanitarian Conference for Sustainability (IIHC), Bengaluru, India, 2022, pp. 204-209, doi: 10.1109/IIHC55949.2022.10060397.

[19] O. H. Alhazmi and Y. K. Malaiya, "Evaluating disaster recovery plans using the cloud," 2013 Proceedings Annual Reliability and Maintainability Symposium (RAMS), Orlando, FL, USA, 2013, pp. 1-6, doi: 10.1109/RAMS.2013.6517700.

[20] A. Arul Mary and K. Chitra, “OGSO-DR: oppositional group search optimizer based efficient disaster recovery in a cloud environment,” Journal of ambient intelligence and humanized computing, vol. 10, no. 5, pp. 1885–1895, 2019.

[21] S. Murthy, “CRYPTOGRAPHIC SECURE CLOUD STORAGE MODEL WITH ANONYMOUS AUTHENTICATION AND AUTOMATIC FILE RECOVERY,” ICTACT Journal on Soft Computing, vol. 05, no. 01, pp. 844–849, Oct. 2014, doi: 10.21917/ijsc.2014.0120.

[22] S. Ruj, M. Stojmenovic and A. Nayak, “Decentralized Access Control with Anonymous Authentication of DataStored in Clouds”, IEEE Transactions on Parallel and Distributed Systems, Vol. 25, No. 2, pp. 556-563, 2013.

[23] M. Raje and D. Mukhopadhyay, "Algorithm for Back-Up and Recovery of Data Stored on Cloud along with Authentication of the User," 2015 International Conference on Information Technology (ICIT), Bhubaneswar, India, 2015, pp. 175-180, doi: 10.1109/ICIT.2015.16.

[24] A. Rahumed, H. C. H. Chen, Y. Tang, P. P. C. Lee and J. C. S. Lui, "A Secure Cloud Backup System with Assured Deletion and Version Control," 2011 40th International Conference on Parallel Processing Workshops, Taipei, Taiwan, 2011, pp. 160-167, doi: 10.1109/ICPPW.2011.17.

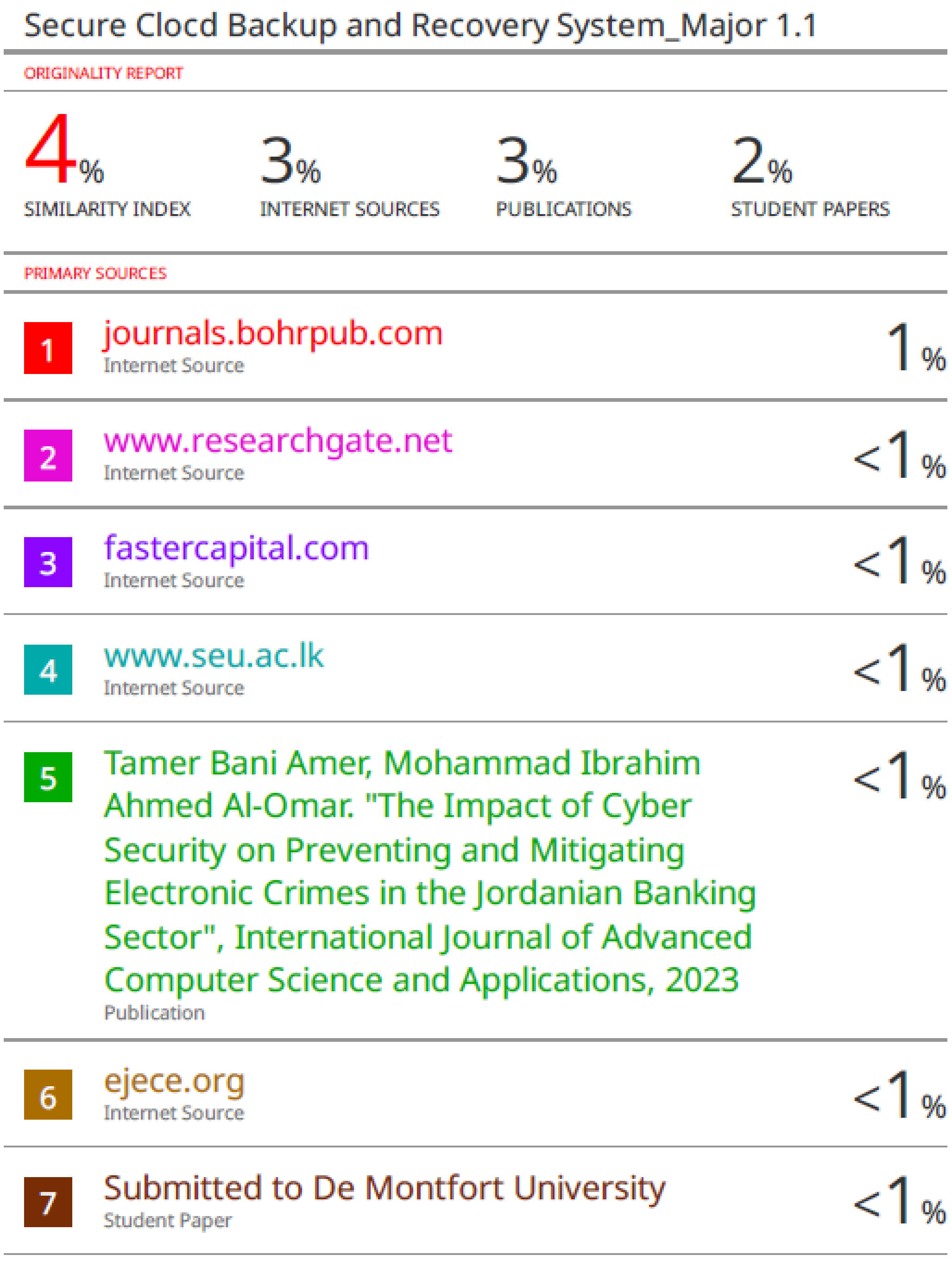
[25] S. Dash and S. K. Pani, “E-Governance Paradigm Using Cloud Infrastructure: Benefits and Challenges,” *Procedia Computer Science*, vol. 85, pp. 843–855, Jan. 2016, doi: 10.1016/j.procs.2016.05.274.

[26] X. Zhang, H. -t. Du, J. -q. Chen, Y. Lin and L. -j. Zeng, "Ensure Data Security in Cloud Storage," 2011 International Conference on Network Computing and Information Security, Guilin, China, 2011, pp. 284-287, doi: 10.1109/NCIS.2011.64.

[27] Cong Wang, Qian Wang, Kui Ren and Wenjing Lou, "Ensuring data storage security in Cloud Computing," 2009 17th International Workshop on Quality of Service, Charleston, SC, 2009, pp. 1-9, doi: 10.1109/IWQoS.2009.5201385.

APPENDIX

PLAGIARISM CERTIFICATE



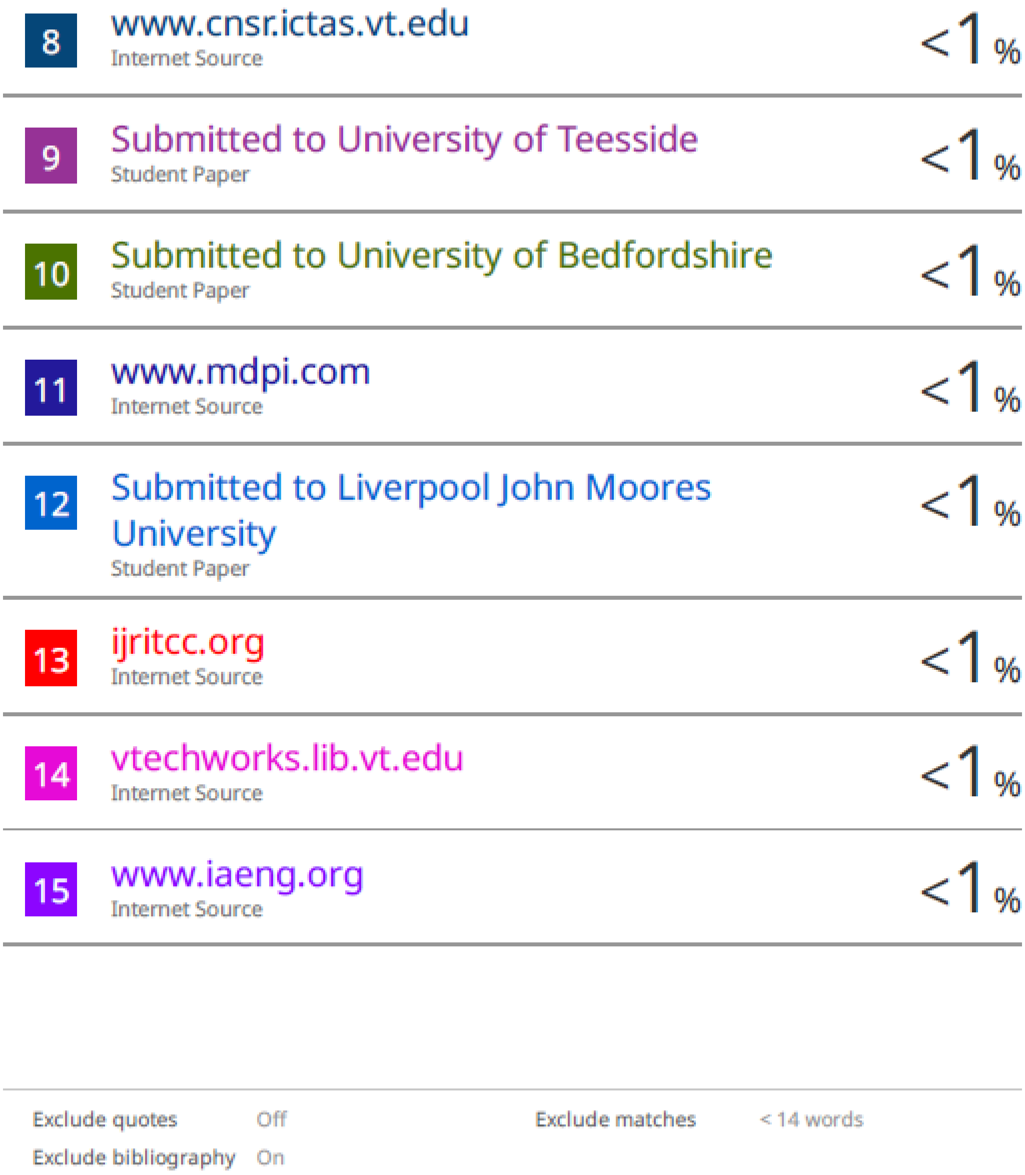
Secure Clocd Backup and Recovery System_Major 1.1

ORIGINALITY REPORT

4%	3%	3%	2%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

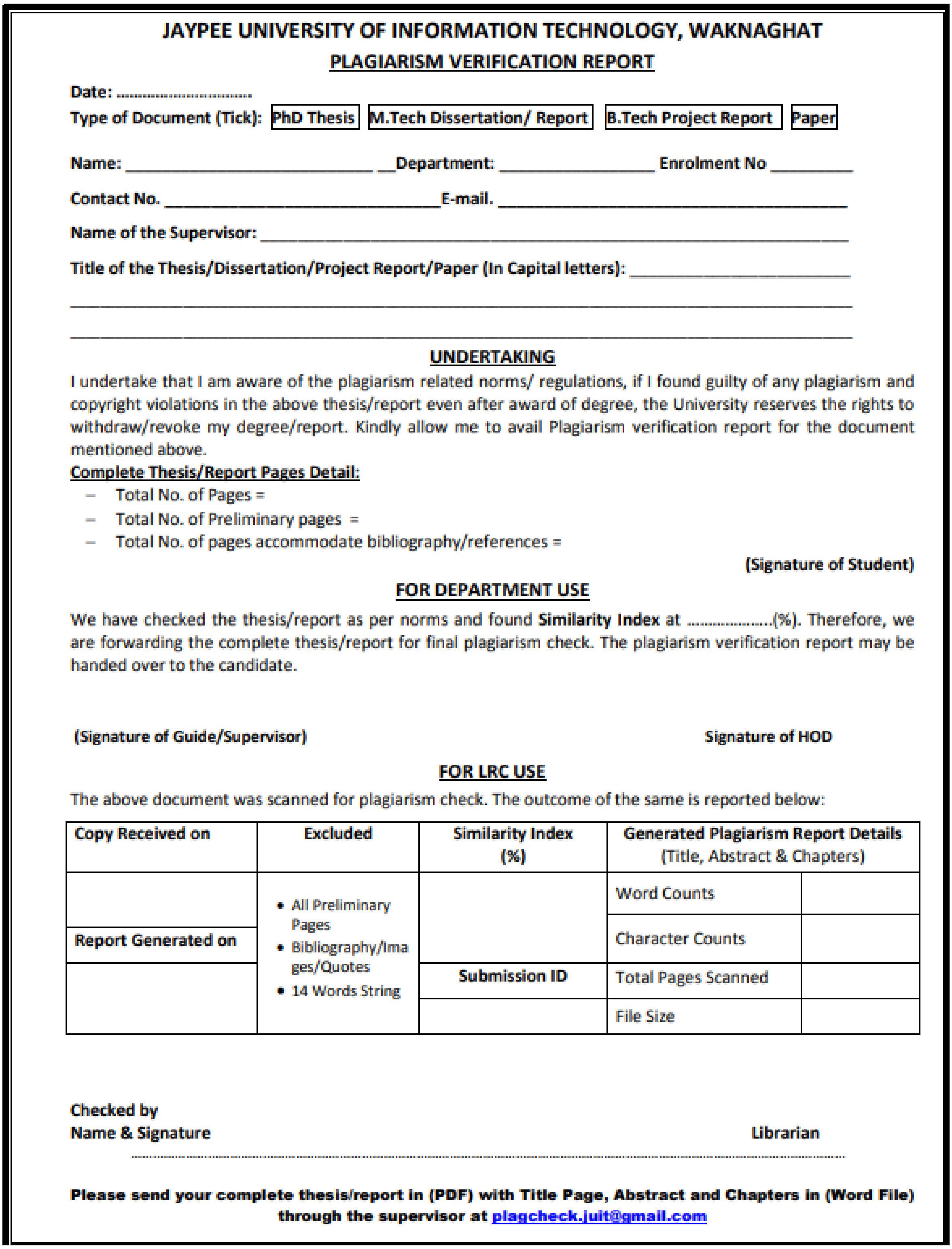
PRIMARY SOURCES

1	journals.bohrpub.com Internet Source	1%
2	www.researchgate.net Internet Source	<1%
3	fastercapital.com Internet Source	<1%
4	www.seu.ac.lk Internet Source	<1%
5	Tamer Bani Amer, Mohammad Ibrahim Ahmed Al-Omar. "The Impact of Cyber Security on Preventing and Mitigating Electronic Crimes in the Jordanian Banking Sector", International Journal of Advanced Computer Science and Applications, 2023 Publication	<1%
6	ejece.org Internet Source	<1%
7	Submitted to De Montfort University Student Paper	<1%



8	www.cnsr.ictas.vt.edu Internet Source	<1 %
9	Submitted to University of Teesside Student Paper	<1 %
10	Submitted to University of Bedfordshire Student Paper	<1 %
11	www.mdpi.com Internet Source	<1 %
12	Submitted to Liverpool John Moores University Student Paper	<1 %
13	ijritcc.org Internet Source	<1 %
14	vtechworks.lib.vt.edu Internet Source	<1 %
15	www.iaeng.org Internet Source	<1 %

Exclude quotes	Off	Exclude matches	< 14 words
Exclude bibliography	On		



JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

PLAGIARISM VERIFICATION REPORT

Date: ....

Type of Document (Tick): PhD Thesis | M.Tech Dissertation/ Report | B.Tech Project Report | Paper

Name: ___________________________ __**Department:** _________________ **Enrolment No** _________

Contact No. ______________________________**E-mail.** ______________________________________

Name of the Supervisor: ____

Title of the Thesis/Dissertation/Project Report/Paper (In Capital letters): ________________________

____

____

UNDERTAKING

I undertake that I am aware of the plagiarism related norms/ regulations, if I found guilty of any plagiarism and copyright violations in the above thesis/report even after award of degree, the University reserves the rights to withdraw/revoke my degree/report. Kindly allow me to avail Plagiarism verification report for the document mentioned above.

Complete Thesis/Report Pages Detail:

- Total No. of Pages =
- Total No. of Preliminary pages =
- Total No. of pages accommodate bibliography/references =

(Signature of Student)

FOR DEPARTMENT USE

We have checked the thesis/report as per norms and found **Similarity Index** at ....(%). Therefore, we are forwarding the complete thesis/report for final plagiarism check. The plagiarism verification report may be handed over to the candidate.

(Signature of Guide/Supervisor) **Signature of HOD**

FOR LRC USE

The above document was scanned for plagiarism check. The outcome of the same is reported below:

Copy Received on	**Excluded**	**Similarity Index (%)**	**Generated Plagiarism Report Details** (Title, Abstract & Chapters)	
	• All Preliminary Pages • Bibliography/Images/Quotes • 14 Words String		Word Counts	
Report Generated on			Character Counts	
		Submission ID	Total Pages Scanned	
			File Size	

Checked by
Name & Signature **Librarian**

....

Please send your complete thesis/report in (PDF) with Title Page, Abstract and Chapters in (Word File) through the supervisor at plagcheck.juit@gmail.com