A MINI PROJECT REPORT

ON

STRENGTHENING CLOUD COMPUTING SECURITY MECHANISMS FOR SECURE KEYWORD SEARCH AND DATA SHARING

Submitted to

SRI INDU COLLEGE OF ENGINEERING AND TECHNOLOGY

In partial fulfillment of the requirements for the award of degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

Submitted By

B.DAYAKAR	[22D41A0511]
B.KAVYA SREE	[22D41A0539]
CH.AKSHITHA	[22D41A0548]
D.MITHIN	[22D41A0552]

Under the guidance of

Mr. P. SUDHEER KUMAR

(Assistant Professor)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SRI INDU COLLEGE OFENGINEERING AND TECHNOLOGY

(An Autonomous Institution under UGC, accredited by NBA, Affiliated to JNTUH)
Sheriguda(V), Ibrahimpatnam (M), Rangareddy Dist – 501 510

(2024 – 2025)

SRI INDU COLLEGE OF ENGINEERING AND TECHNOLOGY

(An Autonomous Institution under UGC, Accredited by NBA, Affiliated to JNTUH)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

Certified that the mini project entitled "**STRENGTHENING CLOUD COMPUTING SECURITY MECHANISMS FOR SECURE KEYWORD SEARCH AND DATA SHARING**" is a Bonafide work carried out by **B.DAYAKAR (22D41A0511), B.KAVYA SREE (22D41A0539), CH.AKSHITHA (22D41A0548), D.MITHIN (22D41A0552)** in partial fulfillment for the award of **Bachelor of Technology** in **Computer Science and Engineering** of SICET, Hyderabad for the academic year 2024-2025.The Project has been approved as it satisfies academic requirements in respect of the work prescribed for **III YEAR, II-SEMESTER** of **B.TECH** course.

INTERNAL GUIDE	**HEAD OF THE DEPARTMENT**
(**Mr. P. SUDHEER KUMAR**) (Assistant Professor)	(**Prof. CH. GVN. PRASAD**)

EXTERNAL EXAMINER

ACKNOWLEDGMENT

The satisfaction that accompanies the successful completion of the task would be put incomplete without the mention of the people who made it possible, whose constant guidance and encouragement crown all the efforts with success.

We are thankful to principal **Prof. Dr. G. Suresh**, for giving us the permission to carry out this project and for providing necessary infrastructure and labs.

We are highly indebted to **Prof. CH. GVN. Prasad**, Head of the Department of Computer Science & Engineering, for his constant motivation, valuable guidance, and support throughout the execution of this project. His insights at every stage greatly contributed to the successful completion of our work.

We would like to thank the Teaching & Non-Teaching staff of Department of Computer Science & Engineering for sharing their knowledge with us.

Last but not the least we express our sincere thanks to everyone who helped directly or indirectly for the completion of this project.

B.DAYAKAR	[22D41A0511]
B.KAVYA SREE	[22D41A0539]
CH.AKSHIHTA	[22D41A0548]
D.MITHIN	[22D41A0552]

INDEX

TOPICS **Page No's**

LIST OF FIGURES

STRENGTHENING CLOUD COMPUTING SECURITY MECHANISMS FOR SECURE KEYWORD SEARCH AND DATA SHARING

ABSTRACT

Cloud computing offers scalable and cost-effective data storage but introduces critical challenges related to data security and privacy. Traditional Attribute-Based Encryption (ABE) provides confidentiality but lacks flexibility for efficient keyword search and secure data sharing. This project presents a Ciphertext-Policy Attribute-Based Keyword Search and Data Sharing (CPAB-KSDS) mechanism that allows users to perform keyword-based search and controlled sharing of encrypted data without requiring decryption or relying on a central Private Key Generator (PKG). The system, developed using Java, AES-256, and RSA-2048, ensures end-to-end encryption, secure key exchange, and searchable encryption through hashed keyword indexing. It supports role-based access control, efficient encryption workflows, and RSA-encrypted symmetric key sharing. By addressing the limitations of conventional ABE systems, the proposed solution enhances usability, scalability, and security, making it suitable for sensitive domains such as healthcare, defense, and legal services. The implementation is proven secure against chosen ciphertext and keyword attacks and demonstrates practical efficiency for real-world deployment.

1. INTRODUCTION

The project "Secure Keyword Search and Data Sharing Mechanism for Cloud Computing" aims to develop a robust system that ensures the security and privacy of sensitive data stored in cloud environments. By implementing advanced encryption, access control, and homomorphic computation techniques, the project seeks to enable authorized users to securely search for specific keywords within their encrypted data while allowing controlled sharing of encrypted content with selected parties, all while maintaining data confidentiality, integrity, and compliance with privacy regulations.

Cloud computing is the use of computing resources (hardware and software) that are delivered as a service over a network (typically the Internet). The name comes from the common use of a cloud shaped symbol as an abstraction for the complex infrastructure it contains in system diagrams. Cloud computing entrusts remote services with a user's data, software and computation. Cloud computing consists of hardware and software resources made available on the Internet as managed third-party services. These services typically provide access to advanced software applications and high-end networks of server computers.

The cloud computing uses networks of large groups of servers typically running low-cost consumer PC technology with specialized connections to spread data-processing chores across them. This shared IT infrastructure contains large pools of systems that are linked together. Often, virtualization techniques are used to maximize the power of cloud computing.

All the existing cloud servers try to store the data in a plain text manner rather than in a encrypted manner. If the encryption key is exposed, the data can be easily accessed by the intruder. In the existing system there is no security for the data even it is encrypted because there is only single cipher key is generated for that data.

As organizations increasingly migrate their data and operations to cloud computing environments, ensuring robust security mechanisms becomes paramount to safeguard sensitive information and maintain the trust of users. Among the critical aspects of cloud security, implementing measures for secure keyword search and data sharing is crucial. This involves protecting data confidentiality, integrity, and accessibility, especially when dealing with sensitive information. In this context, the following strategies are proposed to strengthen cloud computing security mechanisms, with a focus on enhancing the security of keyword searches and facilitating secure data sharing. These strategies encompass encryption, access controls, secure search methods, and proactive monitoring, aiming to create a resilient and trustworthy cloud environment for both enterprises and individual users.

The adoption of cloud computing has revolutionized the way organizations manage and access data, offering unparalleled scalability and flexibility. However, the benefits of cloud computing come with the responsibility to fortify security measures, particularly concerning sensitive data, secure keyword searches, and seamless data sharing. In this era of constant cyber threats, ensuring the confidentiality and integrity of data is critical. This introduction outlines strategies to enhance cloud computing security mechanisms specifically tailored for secure keyword searches and data sharing.

SCOPE OF THE PROJECT

The scope of the project encompasses the design and implementation of a comprehensive system for secure keyword-based search and controlled data sharing within cloud computing environments. This includes developing advanced encryption mechanisms, access control features, and efficient indexing techniques to enable authorized users to search encrypted data while maintaining confidentiality, and allowing data owners to selectively share encrypted content with specific entities. The project aims to provide a user-friendly interface, ensure data integrity, scalability, regulatory compliance, and comprehensive documentation, ultimately enhancing the security and privacy of sensitive data stored and shared in cloud computing.

The scope of a project focused on strengthening cloud computing security mechanisms for secure keyword search and data sharing is comprehensive, encompassing various aspects of cloud security, cryptography, access controls, and user interactions. Here's a breakdown of the potential scope for such a project.

2. LITERATURE SURVEY

TITLE: "Low-cost RF based online patient monitoring using web and mobile applications”.

AUTHORS: V. Goyal, O. Pandey, A. Sahai, and B. Waters

ABSTRACT: As more sensitive data is shared and stored by third-party sites on the Internet, there will be a need to encrypt data stored at these sites. One drawback of encryptingdata, is that it can be selectively shared only at a coarse-grained level (i.e., giving another party your private key). We develop a new cryptosystem for fine-grained sharing of encrypted data that we call Key-Policy Attribute-Based Encryption (KP-ABE). In our cryptosystem, ciphertexts are labelled with sets of attributes and private keys are associated with access structures that control which ciphertexts a user is able to decrypt. We demonstratethe applicability of our construction to sharing of audit-log information and broadcast encryption. Our construction supports delegation of private keys which subsumes Hierarchical Identity-Based Encryption (HIBE).

TITLE: "Ciphertext-Policy Attribute-Based Encryption: An Expressive, Efficient and Provably Secure Realization"

AUTHORS: B. Waters

ABSTRACT: We present a new methodology for realizing Ciphertext-Policy Attribute Encryption (CP-ABE) under concrete and noninteractive cryptographic assumptions in the standard model. Our solutions allow any encryptor to specify access control in terms of any access formula over the attributes in the system. In our most efficient system, ciphertext size, encryption, and decryption time scales linearly with the complexity of the access formula.The only previous work to achieve these parameters was limited to a proof in the generic group model. Our first system is proven selectively secure under a assumption that we callthe decisional Parallel Bilinear Diffie-Hellman Exponent (PBDHE) assumption which can be viewed as a generalization of the BDHE assumption. Our next two constructions provide performance tradeoff’s to achieve provable security respectively under the (weaker) decisional Bilinear-Diffie-Hellman Exponent and decisional Bilinear Diffie-Hellman.

TITLE: " New proof methods for attribute-based encryption: Achieving full security through selective techniques "

AUTHORS: A. Lewko and B. Waters

ABSTRACT: We develop a new methodology for utilizing the prior techniques to prove selective security for functional encryption systems as a direct ingredient in devising proofs of full security. This deepens the relationship between the selective and full security models and provides a path for transferring the best qualities of selectively secure systems to fully secure system.

TITLE: "Using Erasure Codes Efficiently for Storage in a Distributed System".

AUTHORS: M. K. Aguilera, R. Janakiraman, and L. Xu Erasure

ABSTRACT:codes provide space- optimal data redundancy to protect against data loss. A common use is to reliably store data ina distributed system, where erasure-coded data are kept in different nodes to tolerate node failures without losing data. In this paper, we propose a new approach to maintain ensure encoded data in a distributed system. The approach allows the use of space efficient k-of-n erasure codes where n and k are large and the overhead n-k is small. Concurrent updates and accesses to data are highly optimized: in common cases, they require no locks, no two-phase commits, and no logs of old versions of data. We evaluate our approach using an implementation and simulations for larger systems.

TITLE: "Secret-Sharing Schemes: A Survey,"

AUTHORS: Amos beimel.

ABSTRACT: A. Beimel, A secret-sharing scheme is a method by which a dealer distributes shares to parties such that only authorized subsets of parties can reconstruct the secret. Secret-sharing schemes are important tools in cryptography and they are used as a building box in many secure protocols, e.g., general protocol for multiparty computation, Byzantine agreement, threshold cryptography, access control, attribute-based encryption, and generalized oblivious transfer. In this survey, we will describe the most important constructions of secret-sharing schemes, explaining the connections between secret-sharing schemes and monotone formulae and monotone span programs. The main problem with known secret-sharing schemes is the large share size: it is exponential in the number of parties. We conjecture that this is unavoidable. We will discuss the known lower bounds on the share size.

3. SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

The traditional attribute-based encryption is not flexible for data searching and sharing. Additionally, attribute-based encryption is not well scaled when there is an update request to the keyword. In order to search and share a specific record, Alice downloads and decrypts the cipher texts. However, this process is impractical to Alice especially when there are a tremendous number of cipher texts. The worse situation is the data owner Alice should stay online all the time because Alice needs to provide her private key for the data decryption. Thus, ABE solution does not take the advantages of cloud computing.

An alternative method is to delegate a third party to do the search, re-encrypt and keyword update work instead of Alice. Alice can store her private key in the third party's storage, and thus the third party can do the heavy job on behalf of Alice. In such an approach, however, we need to fully trust the third party since it can access to Alice's private key. If the third party is compromised, all the user data including sensitive privacy will be leaked as well. It would be a severe disaster to the users.

All the existing cloud servers try to store the data in a plain text manner rather than in a encrypted manner. If the encryption key is exposed, the data can be easily accessed by the intruder. In the existing system there is no security for the data even it is encrypted because there is only single cipher key is generated for that data.

The existing system refers to the current state of the cloud computing infrastructure, including the technologies, processes, and security measures in place for keyword search and data sharing. Understanding the strengths and weaknesses of the existing system is crucial for planning and implementing improvements. Here's an overview of elements commonly found in an existing cloud computing system:

DISADAVANTAGES

- Many existing systems may not adequately address data privacy concerns, leaving data vulnerable unauthorized access or leakage during keyword search and sharing operations.

- Ensuring the integrity of data stored in the cloud and shared among users is essential. However, existing systems may lack robust mechanisms to detect and prevent unauthorized modifications or tampering of data.
- Some systems may face scalability issues when dealing with large volumes of data or a high number of concurrent users, leading to performance bottlenecks or increased latency.
- Implementing strong security mechanisms often adds complexity to the system, making it more challenging to manage and maintain. This complexity can increase the likelihood of vulnerabilities or misconfigurations that could be exploited by attackers.
- Effective key management is critical for ensuring the security of encrypted data in the cloud. However, existing systems may struggle with key management issues, such as key distribution, rotation, and revocation, leading to potential security weaknesses.

3.2 PROPOSED SYSTEM

Prior work did not demonstrate that the existing attribute-based mechanisms could both support keyword search and data sharing in one scheme without resorting to PKG. Therefore, a new attribute-based mechanism is needed to achieve the goal for the above PHR scenario. One may argue that the problem can be trivially solved by combining an ABPRE scheme and attribute-based keyword search scheme (AB-KS). However, the combination could result in two major issues: 1) the combined scheme is not CCA secure, 2) it is vulnerable to collusion attack.

Therefore, a secure scheme is desired to fully support keyword searching, data sharing as well as the protection of the privacy of keyword. All of these concerns motivate us to design a mechanism that:

1) Allows the data owner to search and share the encrypted health report without the unnecessary decryption process.

2) Supports keyword updating during the data sharing phase.

3) More importantly, does not need the exist of the PKG, either in the phase of data sharing or keyword updating.

4) The data owner can fully decide who could access the data he encrypted.

We first introduce a cipher text-policy attribute-based mechanism with keyword search and data sharing (CPAB-KSDS) for encrypted cloud data. The searching and sharing functionality are enabled in the cipher text-policy setting. Furthermore, our scheme supports the keyword to be updated during the sharing phase. After presenting the construction of our mechanism, we proof its chosen cipher text attack (CCA) and chosen keyword attack (CKA) security in the random oracle model. The proposed construction is demonstrated practical and efficient in the performance and property comparison.

In this proposed system, we study data confidentiality against an adversary which knows the encryption key and has access to a large fraction of the cipher text blocks. The adversary can acquire the key either by exploiting flaws or backdoors in the key-generation software, or by compromising the devices that store the keys (e.g., at the user-side or in the cloud). To counter such an adversary, we propose a novel and efficient scheme which ensures that plaintext data cannot be recovered as long as the adversary has access to at most all but two ciphertext blocks, even when the encryption key is exposed.

ADVANTAGES

1. We describe the notion of CPAB-KSDS as well as its security model.
2. The proposed construction is demonstrated practical and efficient in the performance and property comparison a Ciphertext-Policy Attribute-Based Encryption scheme that is proven fully secure while matching the efficiency of the state of the art selectively secure systems

3. Here we use multiple cipher text keys for decrypting the data 2. It is very hard for the user to break the keys.

4. The proposed system is really complex and tough to break the encrypted data without having key premises

5. The proposed system for strengthening cloud computing security mechanisms for secure keyword search and data sharing offers several advantages over the existing system. These advantages aim to address the identified weaknesses and enhance overall security. Here are some potential advantages of the proposed system:

6. The proposed system incorporates advanced encryption techniques for both data at rest and in transit, significantly strengthening data security. This ensures that even if unauthorizedaccess occurs, the data remains confidential and protected.

4. SYSTEM REQURIMENTS

4.1 FUNCTIONAL REQUIREMENTS

1. HEALTH RECORD OWNER:
2. DELEGATEE:
3. DELEGATOR:
4. PKG:
5. CLOUD SERVER:

4.2.1 HARDWARE REQUIREMENTS

MINIMUM(Required for Execution)		**MY SYSTEM (Development)**
System	Pentium IV 2.4 GHz	i3 Processor 5th Gen
HardDisk	40 Gb	512 Gb
RAM	1 Gb	4 Gb

4.2.2 SOFTWARE REQUIREMENTS

Operating System	Windows 10/11
Development Software	JAVA 8.1
Programming Language	Java
Front End Technologies	HTML5, CSS3, Java Script
Database Language	SQL
Database (RDBMS)	MySQL
Database Software	MySQL Server
Web Server or Deployment Server	Apache tomcat
Design/Modelling	Rational Rose

5. SYSTEM STUDY

5.1 FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth witha very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understandingof the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

1. ECONOMICAL FEASIBILITY
2. TECHNICAL FEASIBILITY
3. SOCIAL FEASIBILITY

ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. Thus, the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

Economic feasibility refers to the assessment of whether a proposed project or investment is financially viable and justifiable within the constraints of available resources. This evaluation involves analyzing the potential costs and benefits associated with the project over its lifecycle. Factors such as initial investment, operating costs, anticipated revenues, and potential savings are considered to determine if the project will yield a positive return on investment. Economic feasibility studies help decision-makers understand the financial implications of a project, assess its affordability, and make informed choices about resource allocation. Additionally, these studies often consider factors such as market conditions, economic trends, and the project's impact on overall organizational or societal economic well-being.

TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

Economic feasibility refers to the assessment of whether a proposed project or investment is financially viable and justifiable within the constraints of available resources. This evaluation involves analyzing the potential costs and benefits associated with the project over its lifecycle. Factors such as initial investment, operating costs, anticipated revenues, and potential savings are considered to determine if the project will yield a positive return on investment. Economic feasibility studies help decision-makers understand the financial implications of a project, assess its affordability, and make informed choices about resource allocation. Additionally, these studies often consider factors such as market conditions, economic trends, and the project's impact on overall organizational or societal economic well-being. A project is deemed economically feasible if the anticipated financial benefits outweigh the costs, making it a prudent and sustainable investment.

SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by theusers solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also ableto make some constructive criticism, which is welcomed, as he is the final user of the system.

Social feasibility involves evaluating the acceptability and impact of a proposed project within the community or society it aims to serve. This assessment considers the project's alignment with social values, cultural norms, and ethical standards. It examines how the project may affect different demographic groups and stakeholders, ensuring that it promotes inclusivity and does not lead to social inequalities. Social feasibility studies also assess potential social resistance, concerns, or ethical dilemmas that may arise during or after project implementation. A socially feasible project should contribute positively to the community, fostering social cohesion, addressing societal needs, and respecting the rights and interests of diverse stakeholders. This evaluation is integral to ensuring that the project aligns with broader social objectives and does not inadvertently lead to negative social consequences .

5.2 FEASIBILITY ANALYSIS

Technical Feasibility:

Technology Stack: Here's why these three technologies are integral to any web development project: The system's technical prerequisites can be readily fulfilled through the utilization of widely accessible technologies, such as data mining libraries, relational databases, and statistical tools. These foundational components collectively form a robust and accessible framework for developing and operating the system.

Relational databases, exemplified by MySQL, offer scalable and efficient data storage solutions. These databases empower the system to store and manage data in a structured manner, facilitating seamless retrieval and manipulation.

By leveraging these commonplace technologies, the system can harness the full potential of data mining, storage, and analysis, enabling it to operate efficiently and effectively. These readily available resources not only expedite development but also enhance the system's accessibility to a wider audience of developers and practitioners.

Incorporating HTML, CSS, and Java into the tech stack is essential for building modern, interactive, and visually appealing web application.

HTML (Hypertext Markup Language): HTML serves as the structural foundation of web pages. It defines the layout and content structure of your application’s user interface. With HTML, you can create various elements like forms, headings, paragraphs, and links. It is essential for presenting content to users in a structured and meaningful way.

CSS (Cascading Style Sheets): CSS complements HTML by providing styling and design capabilities. It allows you to control the visual presentation of your web application, including aspects such as fonts, colors, layout, and responsiveness. By using CSS, you can ensure that your application is visually appealing, user-friendly, and responsive across different devices and screen sizes.

Java (Backend): Java is a powerful and versatile programming language that is well- suited for building the backend of web applications. It offers strong support for server-side processing, business logic implementation, database connectivity, and handling user requests. Java's scalability, reliability, and robustness make it a popular choice for developing enterprise-grade web applications. Popular Java frameworks like Spring and Java EE provide a structured and efficient way to build the backend of your web application.

By combining HTML, CSS, and Java, we create a full-stack web development environment that covers both the frontend and backend aspects of the system. This combination allows for seamless communication between the user interface and backend logic, enabling the creation of dynamic and interactive web application. Moreover, Java's stability and extensive libraries make it a reliable choice for building the core functionality of our system while HTML and CSS ensure an engaging and visually appealing user experience.

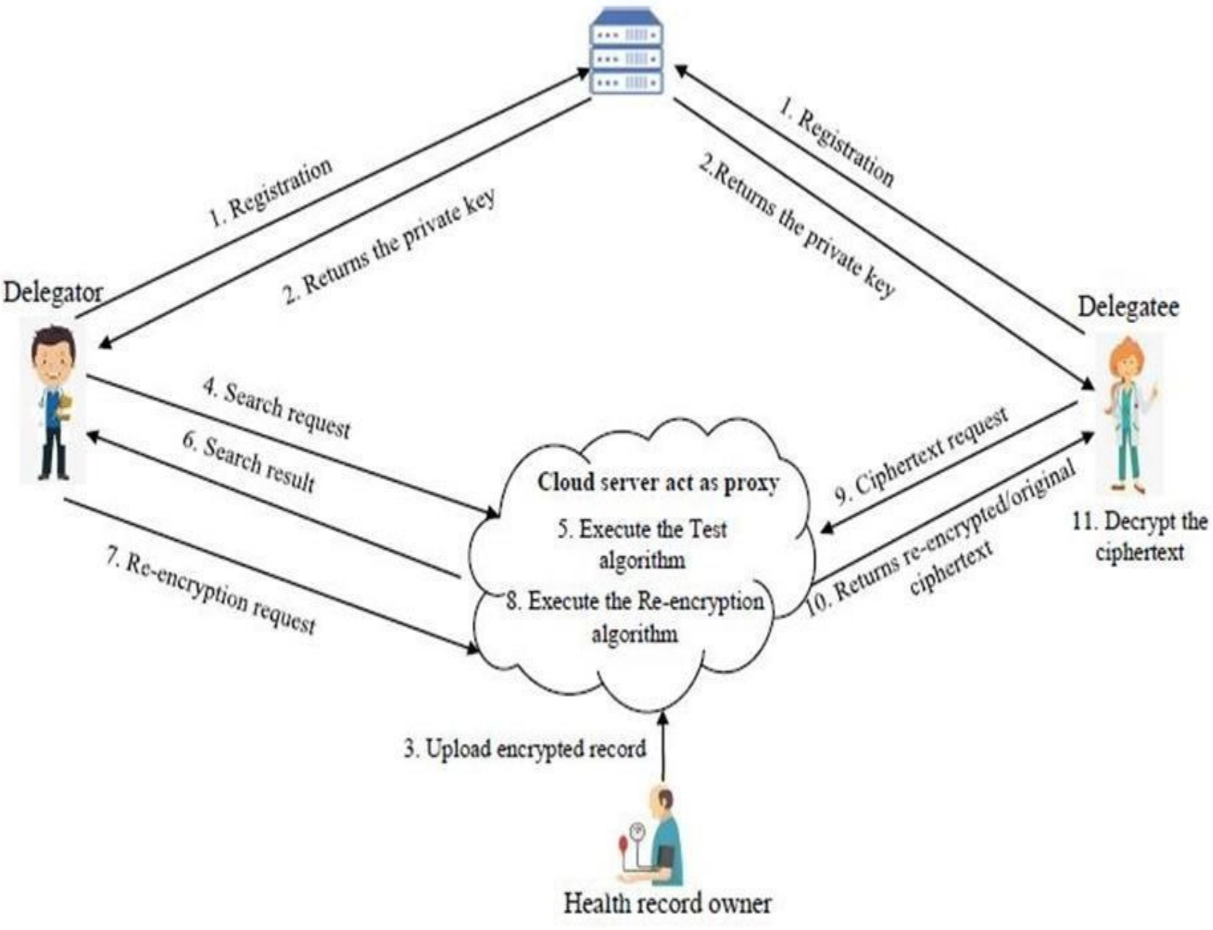
Operational Feasibility:

Data Availability: We can confidently affirm that the essential user behaviour data, inclusive of time distribution parameters, is both accessible and abundant in quantity, ensuring the system's robust operation. The assurance stems from the fact that the necessary data, which encapsulates user actions and their temporal patterns, is readily obtainable. It is obtainable through various means, including user interaction logs, website analytics, and application usage tracking. These sources consistently furnish us with a rich dataset that encapsulates how users engage with the system over time. Moreover, the ample availability of this data serves as a foundation for informed decision-making and data-driven insights. It enables the system to not only monitor user interactions but also to derive meaningful patterns and trends from these interactions. The abundance of data ensures that the system has a substantial historical record to draw upon, facilitating accurate predictions and intelligent responses to user behaviour. This wealth of user behaviour data, along with its time-related parameters, not only meets but exceeds the system's requirements.

Integration: The system's integration feasibility aligns seamlessly with the pre-existing security infrastructure and user monitoring systems in place. This compatibility underscores the system's capacity to harmoniously coexist with the current security and monitoring ecosystem. The design of the system has been meticulously crafted to ensure it adheres to the established security protocols and practices within the organization. This includes robust encryption methods, access controls, and authentication mechanisms, all of which can seamlessly interface with the existing security framework. The system's compliance with industry-standard security measures further underscores its suitability for integration. Additionally, the system's adaptability extends to user monitoring systems. It can seamlessly interface with the organization's user activity tracking and monitoring tools, facilitating the continuous surveillance of user interactions within the system. This ensures that the organization can maintain its vigilant stance on security and user behaviour, leveraging its current monitoring investments effectively. In summary, the system's integration feasibility is underpinned by its innate compatibility with the organization's security infrastructure and user monitoring systems. Its design is rooted in alignment with established security protocols and offers a cohesive fit within the existing technological landscape, thereby streamlining the integration process and bolstering the organization's overall security posture.

6. SYSTEM ARCHITECTURE

6.1 SYSTEM ARCHITECTURE


1. Registration
2. Returns the private key
1. Registration
2.Returns the private key
Delegator
Delegatee
4. Search request
6. Search result
7. Re-encryption request
Cloud server act as proxy
5. Execute the Test algorithm
8. Execute the Re-encryption algorithm
9. Ciphertext request
10. Returns re-encrypted/original ciphertext
11. Decrypt the ciphertext
3. Upload encrypted record
Health record owner

6.2 UML DIAGRAMS

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

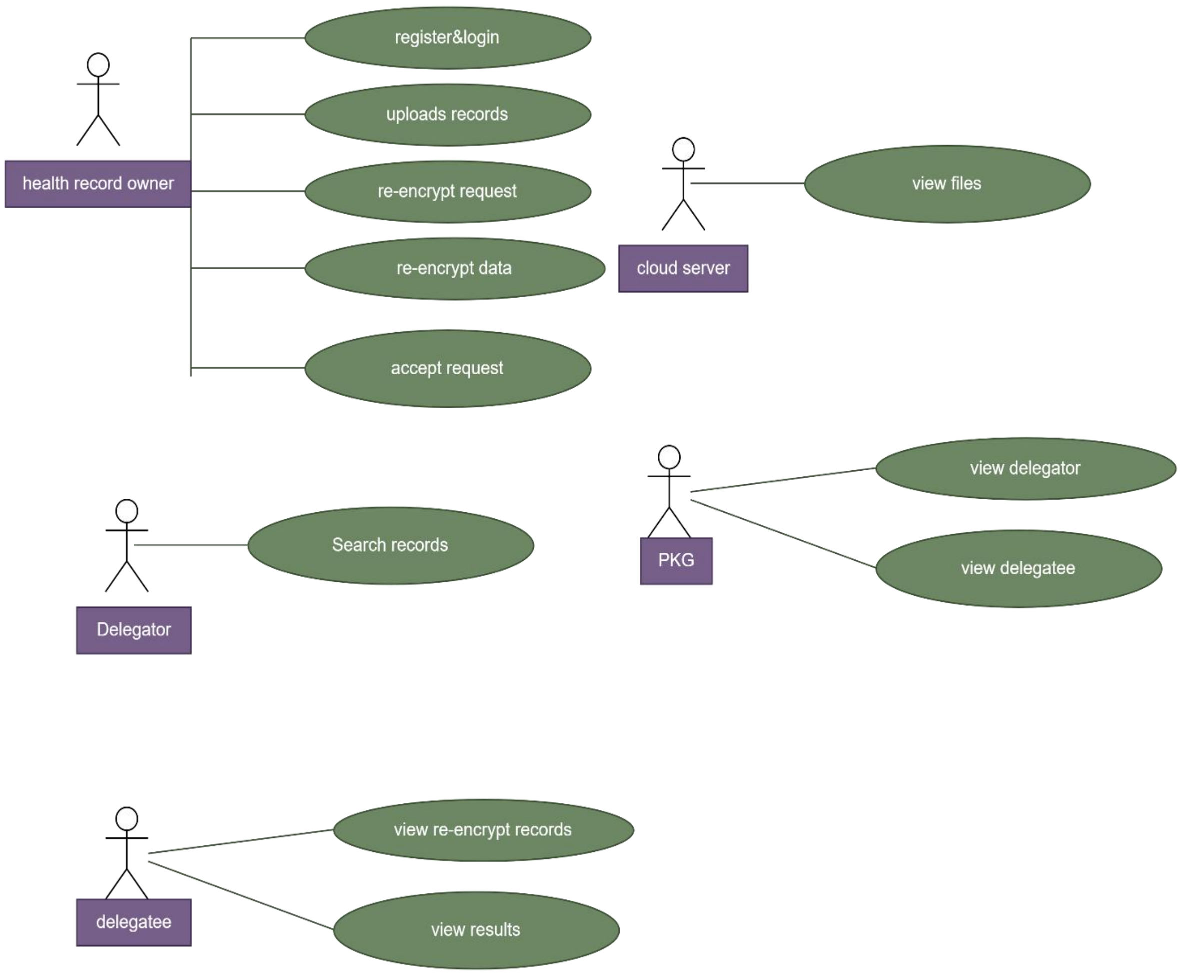
GOALS:

The Primary goals in the design of the UML are as follows:

1. Provide extendibility and specialization mechanisms to extend the core concepts.
2. Be independent of particular programming languages and development process.
3. Provide a formal basis for understanding the modeling language.
4. Encourage the growth of OO tools market.
5. Support higher level development concepts such as collaborations,
6. Integrate best practices.

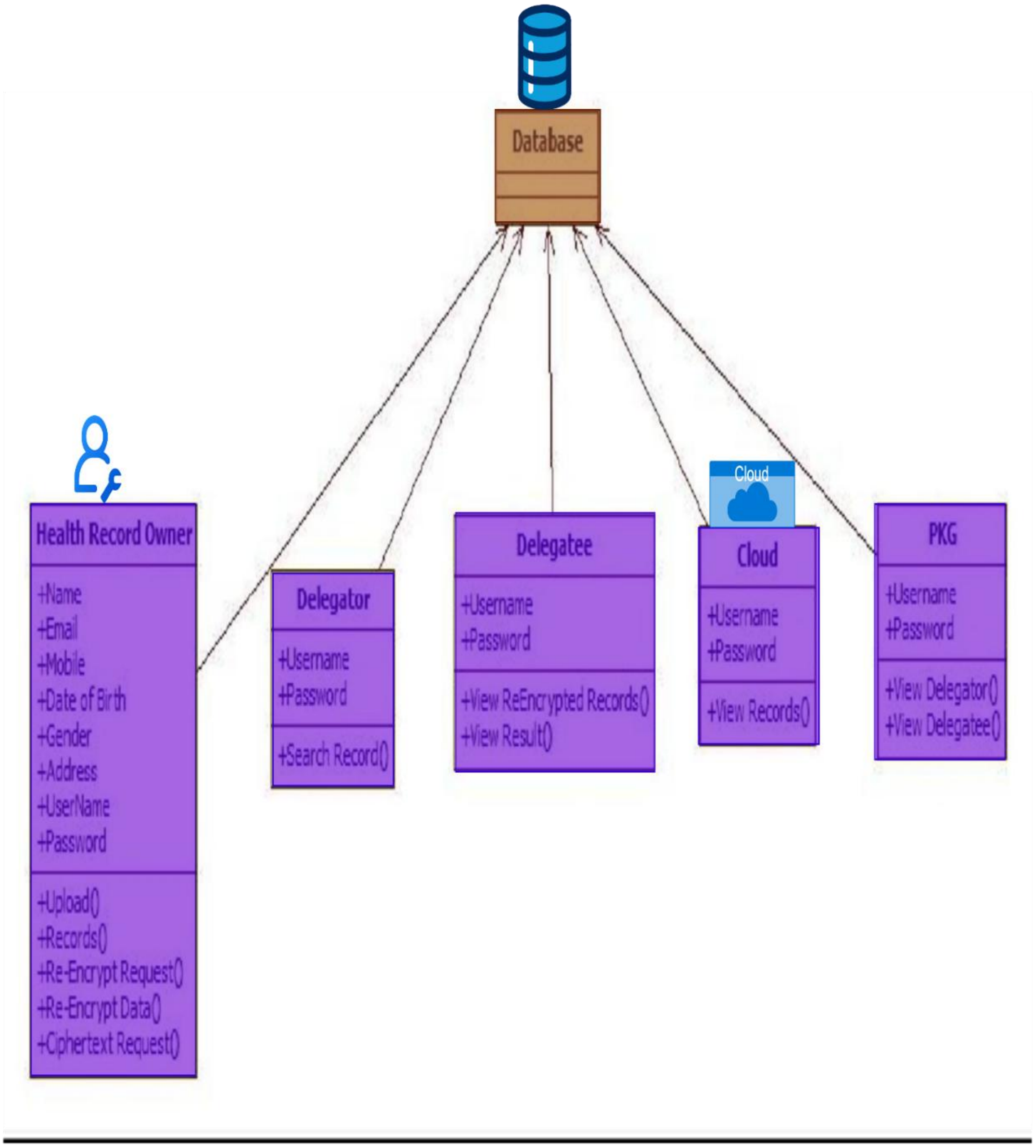
6.2.1 USE CASE DIAGRAM:

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



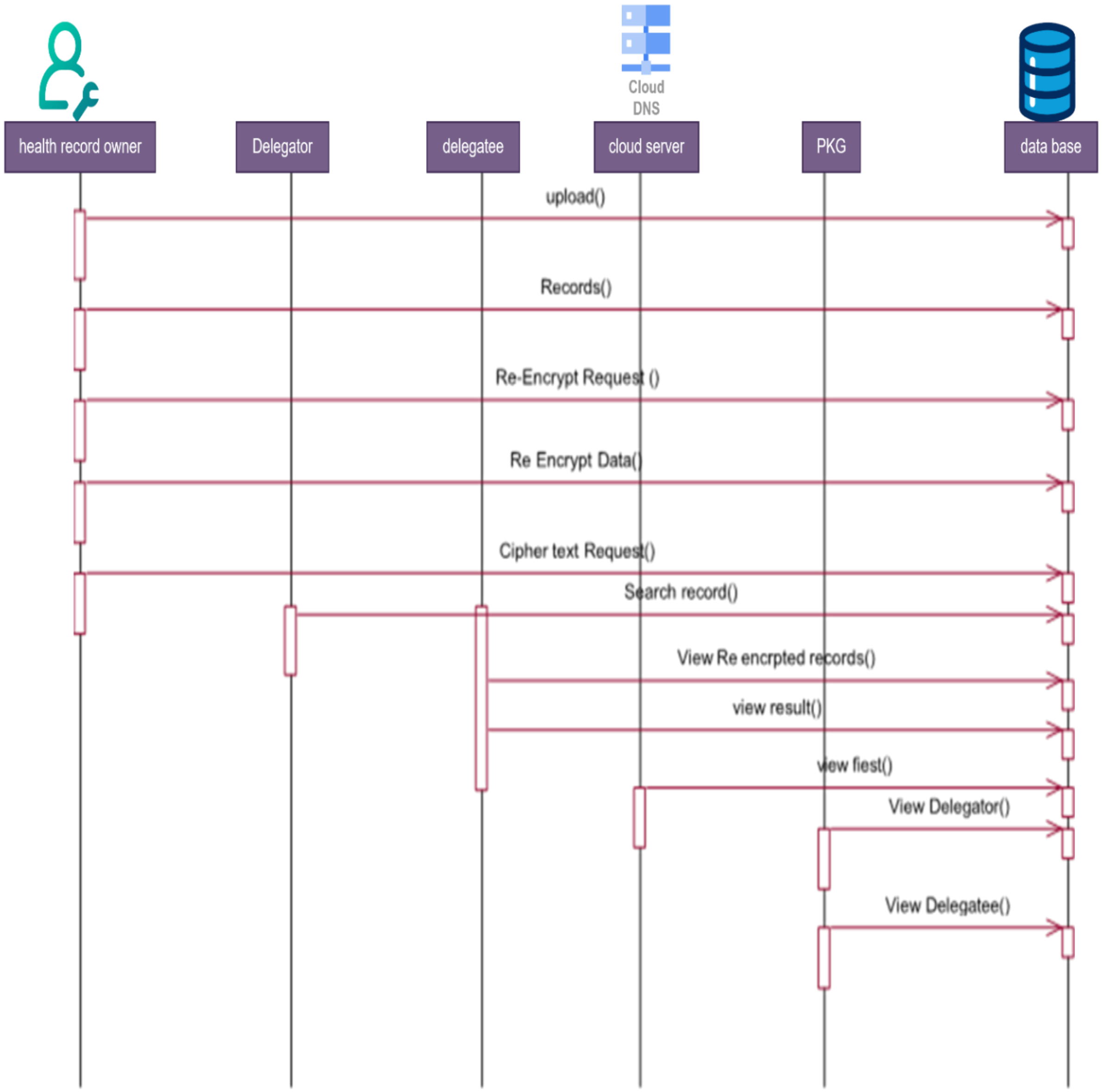
6.2.2 CLASS DIAGRAM:

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



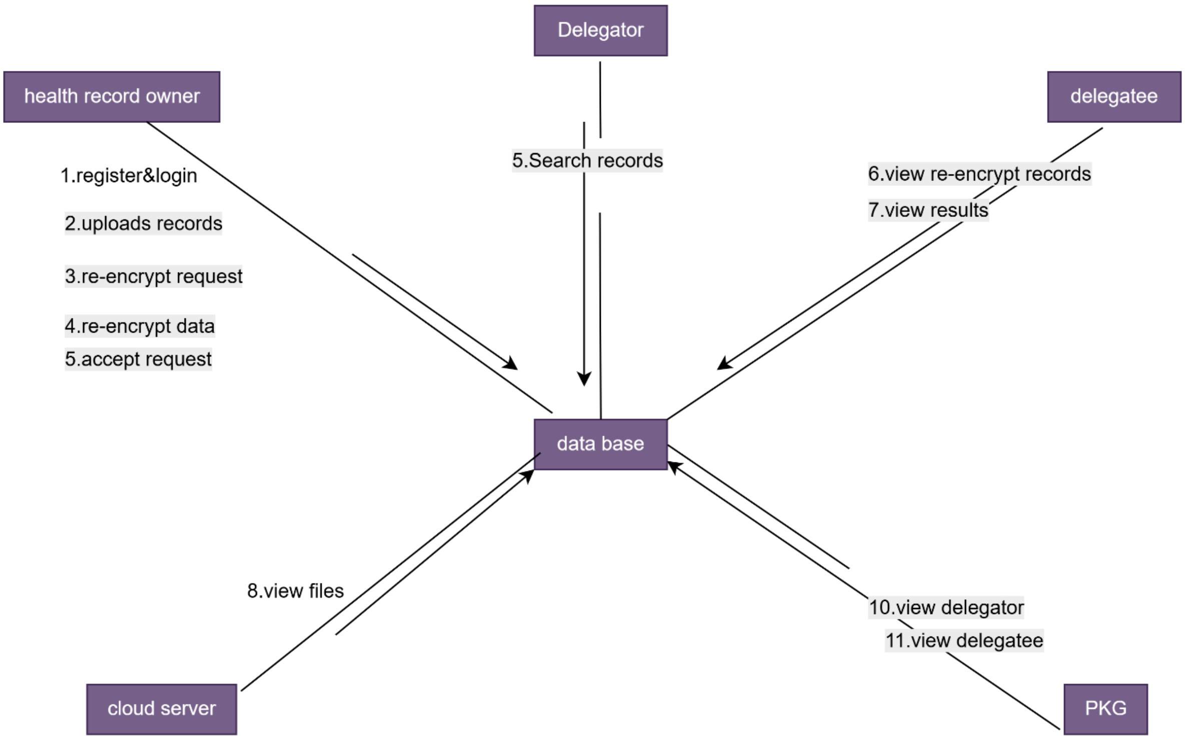
6.2.3 SEQUENCE DIAGRAM:

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



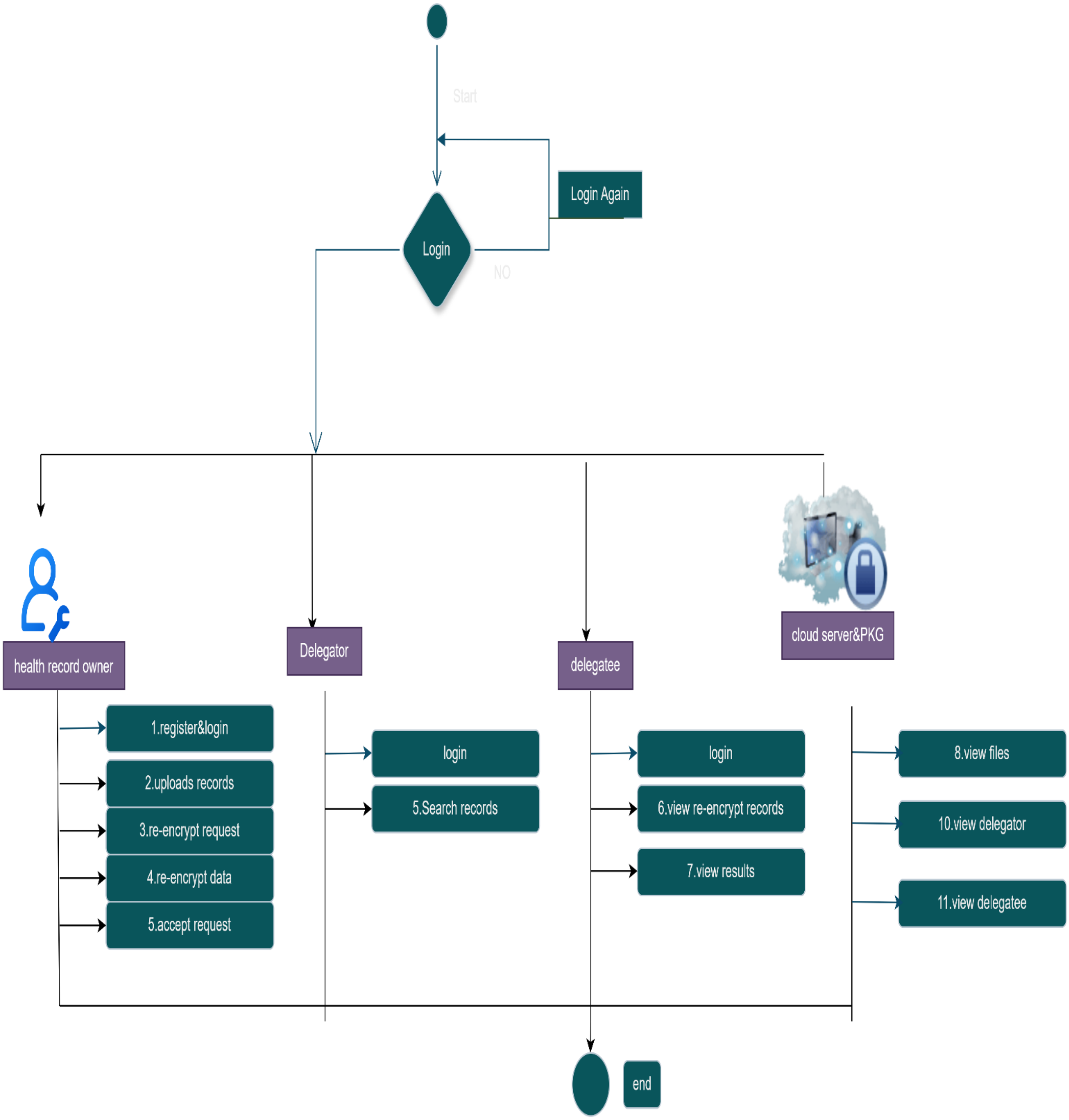
6.2.4 COLLABORATION DIAGRAM

A collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software objects in the Unified Modeling Language (UML). These diagrams can be used to portray the dynamic behavior of a particular usecase and define the role of each object.



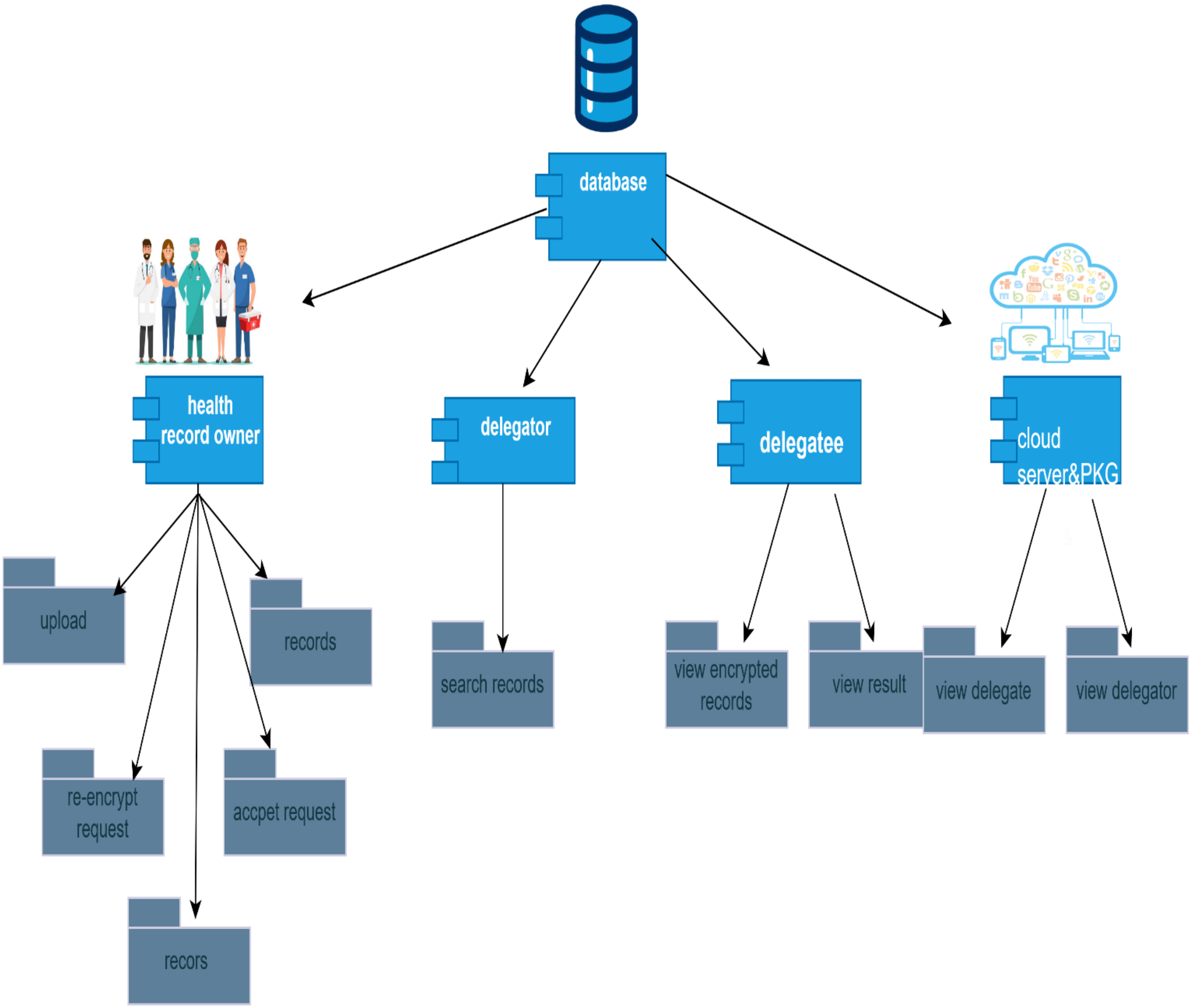
6.2.5 ACTIVITY DIAGRAM

Activity diagrams are similar to flowcharts because they show the flow between the actions in an activity; however, activity diagrams can also show parallel or concurrent flows and alternate flows. In activity diagrams, you use activity nodes and activity edges to model the flow of control and data between actions.



6.2.6 COMPONENT DIAGRAM

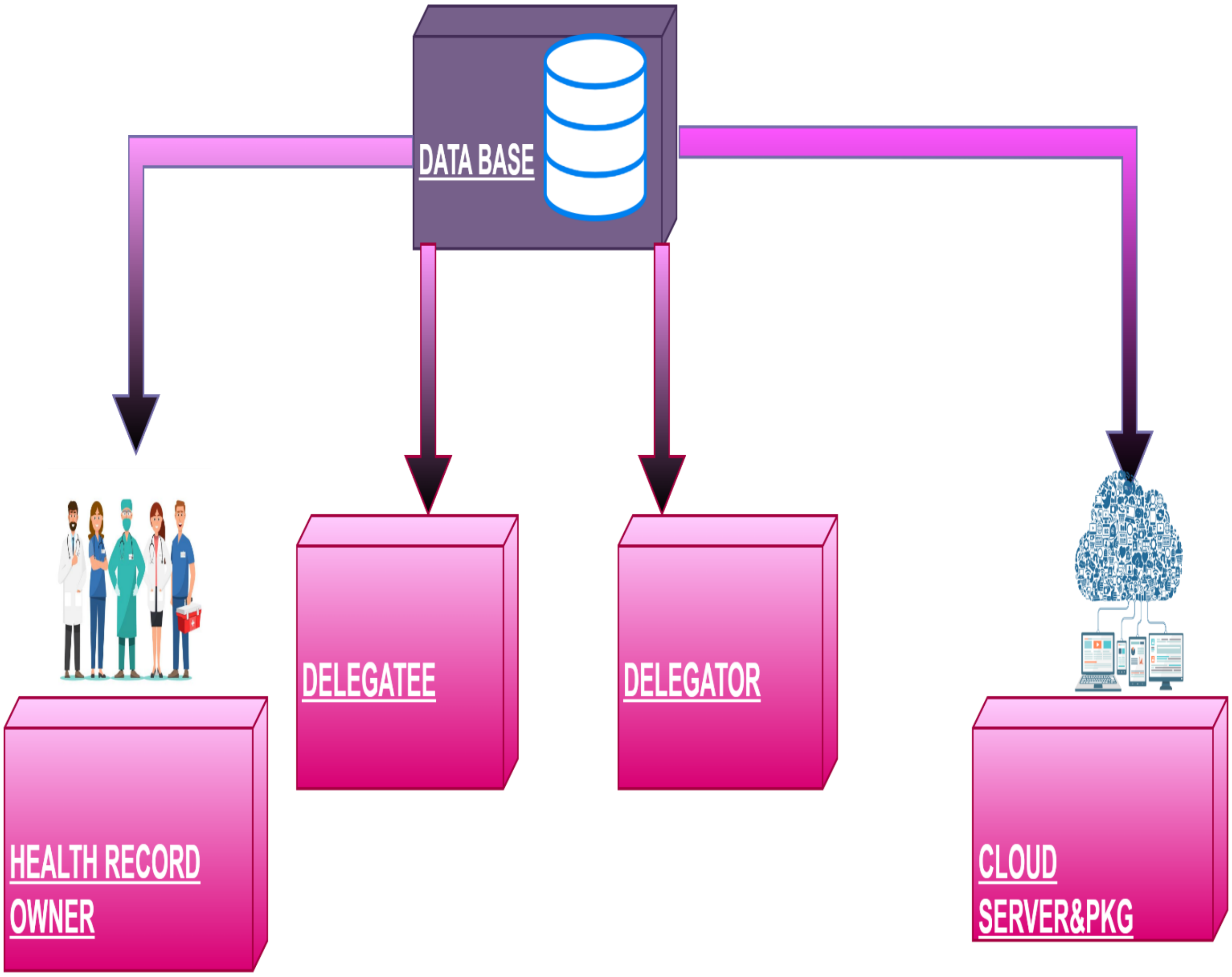
A component diagram is used to break down a large object-oriented system into the smaller components, so as to make them more manageable. It models the physical view of a system such as executables, files, libraries, etc. that resides within the node It visualizes the relationships as well as the organization between the components present in the system. It helps in forming an executable system. A component is a single unit of the system, which is replaceable and executable. The implementation details of a component are hidden, andit necessitates an interface to execute a function. It is like a black box whose behavior is explainedby the provided and required interfaces.



6.2.7 DEPLOYMENT DIAGRAM

A deployment diagram is a UML diagram type that shows the execution architecture of a system, including nodes such as hardware or software execution environments, and the middleware connecting them.

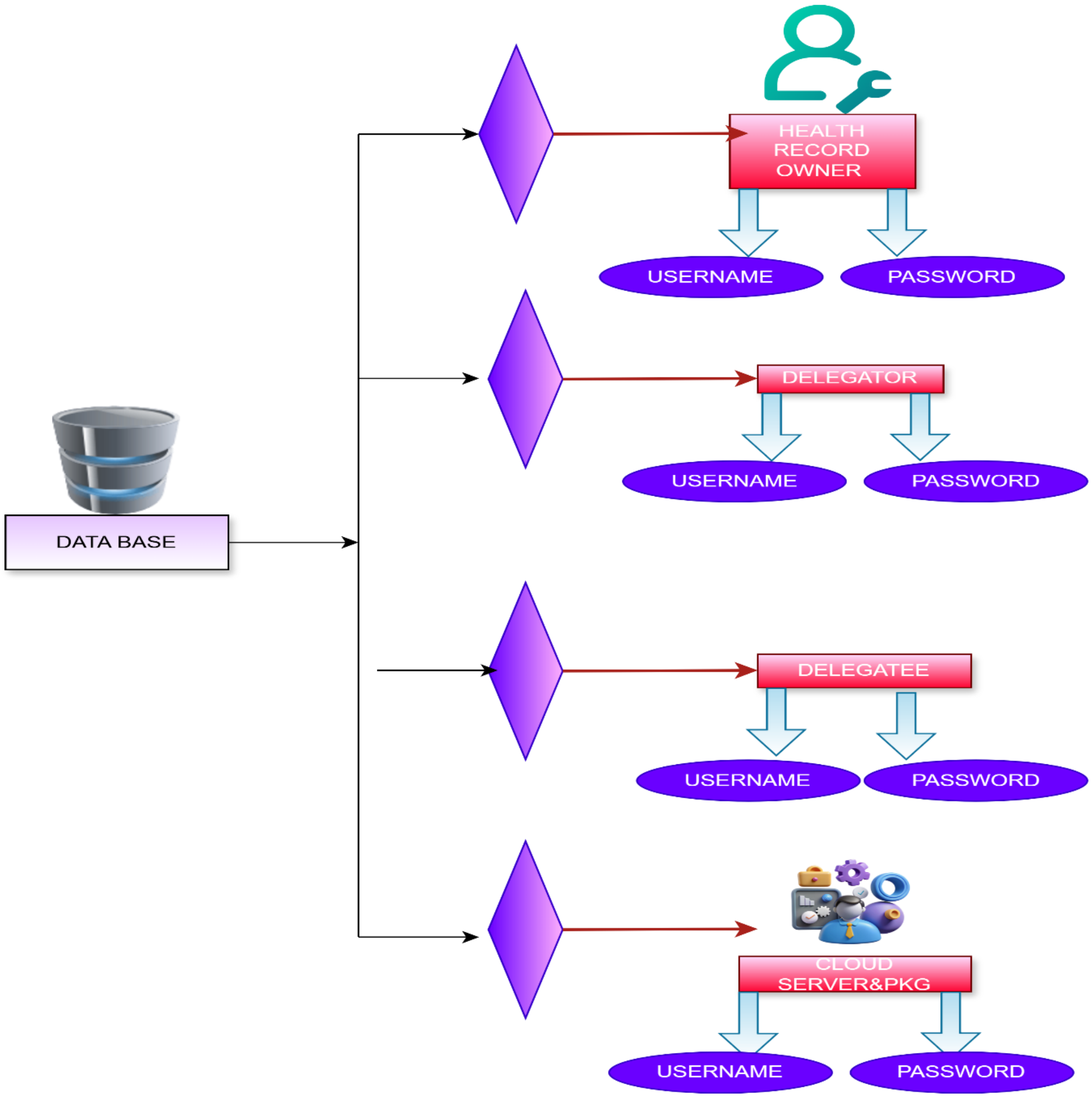
Deployment diagrams are typically used to visualize the physical hardware and software of a system. Using it you can understand how the system will be physically deployed on the hardware. Deployment diagrams help model the hardware topology of a system compared to other UML diagram types which mostly outline the logical components of a system.



6.2.8 ER DIAGRAM

The Entity Relational Model is a model for identifying entities to be represented in the database and representation of how those entities are related. The ER data model specifies enterprise schema that represents the overall logical structure of a database graphically.

The Entity Relationship Diagram explains the relationship among the entities present in the database. ER models are used to model real-world objects like a person, a car, or a company and the relation between these real-world objects. In short, the ER Diagram is the structural format of the database.



6.2.9 DATA DICTIONERY

A data dictionary contains metadata i.e., data about the database. The data dictionary is very important as it contains information such as what is in the database, who is allowed to access it, where is the database physically stored etc. The users of the database normally don't interact with the data dictionary, it is only handled by the database administrators.

USER REGISTRATION TABLE:

FIELD	TYPE	NULL	KEY	DEFAULT	DESCRIPTION
Name	Varchar (20)	NOT NULL	Primary key	NULL	Enter the name
email	Varchar (20)	NOT NULL		NULL	Enter the mail
mobile	int	NOT NULL		NULL	Enter the mobile number
Address	Varchar (20)	NOT NULL		NULL	Enter the Address
username	Varchar (20)	NOT NULL		NULL	Enter the user name
password	Varchar (20)	NOT NULL		NULL	Enter the password

USER LOGIN TABLE:

FIELD	TYPE	NULL	KEY	DEFAULT	DESCRIPTION
username	Varchar(20)	NOT NULL		NULL	Enter the user name
password	Varchar(20)	NOT NULL		NULL	Enter the password

DELEGATEE REGISTRATION TABLE:

FIELD	TYPE	NULL	KEY	DEFAULT	DESCRIPTION
Name	Varchar (20)	NOT NULL	Primary key	NULL	Enter the name
email	Varchar (20)	NOT NULL		NULL	Enter the mail
mobile	int	NOT NULL		NULL	Enter the mobile number
Address	Varchar (20)	NOT NULL		NULL	Enter the Address
username	Varchar (20)	NOT NULL		NULL	Enter the user name
password	Varchar (20)	NOT NULL		NULL	Enter the password

DELEGATEE LOGIN TABLE:

FIELD		NULL	KEY	DEFAULT	DESCRIPTION
username	Varchar (20)	NOT NULL		NULL	Enter the user name
password	Varchar (20)	NOT NULL		NULL	Enter the password

DELEGATEE REGISTRATION TABLE:

FIELD	TYPE	NULL	KEY	DEFAULT	DESCRIPTION
Name	Varchar (20)	NOT NULL	Primary key	NULL	Enter the name
email	Varchar (20)	NOT NULL		NULL	Enter the mail

mobile	int	NOT NULL		NULL	Enter the mobile number
Address	Varchar (20)	NOT NULL		NULL	Enter the Address
username	Varchar (20)	NOT NULL		NULL	Enter the user name
password	Varchar (20)	NOT NULL		NULL	Enter the password

DELEGATEE LOGIN TABLE:

FIELD	TYPE	NULL	KEY	DEFAULT	DESCRIPTION
username	Varchar (20)	NOT NULL		NULL	Enter the user name
password	Varchar (20)	NOT NULL		NULL	Enter the password

PKG LOGIN TABLE:

FIELD	TYPE	NULL	KEY	DEFAULT	DESCRIPTION
username	Varchar (20)	NOT NULL		NULL	Enter the user's name
password	Varchar (20)	NOT NULL		NULL	Enter the password

CLOUD SERVER TABLE:

FIELD	TYPE	NULL	KEY	DEFAULT	DESCRIPTION
username	Varchar (20)	NOT NULL		NULL	Enter the user's name
password	Varchar (20)	NOT NULL		NULL	Enter the password

7. INPUT AND OUTPUT DESIGN

7.1 INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design consideredthe following things:

1. What data should be given as input?
2. How the data should be arranged or coded?
3. The dialog to guide the operating personnel in providing input.
4. Methods for preparing input validations and steps to follow when error occur.

Input design is a crucial phase in the system development life cycle that focuses on defining and specifying how data is entered into a computer system. It is a critical aspect of user interface design and involves determining the methods and mechanisms for collecting and inputting data efficiently and accurately. The objective of input design is to create a user-friendly and error-free interface that aligns with the system's requirements and user expectations. Key considerations in input design include data validation, data verification, data entry methods (such as forms, dialog boxes, or direct data entry), input controls, and error-handling mechanisms. Designers aim to minimize data entry errors, enhance user productivity, and ensure that the input process aligns with the overall goals and functionality of the system. A well-designed input system contributes to the overall efficiency and reliability of the entire information system.

OBJECTIVES

1. Input Design is the process of converting a user-oriented description of the input into a computer- based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.

2. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also providesrecord viewing facilities.

3. When the data is entered it will check for its validity. Data can be entered with the help of screens.

7.2 OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.

2. Select methods for presenting information.

3. Create document, report, or other formats that contain information produced by thesystem.

The output form of an information system should accomplish one or more of the followingobjectives.

1.Convey information about past activities, current status or projections of the Future.

2.Signal important events, opportunities, problems, or warnings.

3.Trigger an action.

4.Confirm an action.

Output design is a critical phase in the system development life cycle that focuses on presenting information to users in a clear, meaningful, and effective manner. The goal of output design is to deliver information to users or other systems in a format that meets their needs, facilitates decision-making, and enhances overall system usability. Key considerations in output design include selecting appropriate output formats (such as reports, graphs, or visual displays), organizing and formatting the information, determining the frequency and timing of output generation, and ensuring that the output aligns with the system's objectives and user requirements. Designers aim to present information in a way that is easily understandable, relevant, and supports users in making informed decisions. A well-designed output system enhances the overall user experience, contributes to system effectiveness, and promotes efficient utilization of information generated by the system.

8. IMPLEMENTATION

8.1 MODULES

Health record owner

Delegatee

Delegator

PKI

Cloud server

8.1.1 MODULE DESCRIPTION

Health record owner

The term "health record owner" typically refers to the individual who is the subject of a health record or a patient's health information. In the context of healthcare and medical records, the owner is the person to whom the health information pertains. Here are some key points to understand::n many jurisdictions and healthcare systems, patients are considered the owners of their health records. This ownership is grounded in the principle of patient autonomy and the right to control one's health information.

Delegatee:

In the context of healthcare, a delegatee is an individual or entity to whom certain responsibilities or tasks related to patient care or health information management are assigned by a healthcare provider or organization. This delegation is typically guided by legal and ethical considerations, ensuring that the delegatee has the appropriate qualifications and training to perform the delegated tasks. For instance, a healthcare provider may delegate specific administrative tasks, such as appointment scheduling or data entry, to support staff within the organization. Delegation can also extend to sharing certain patient information with other healthcare professionals involved in a patient's care, emphasizing the importance of maintaining patient privacy and confidentiality even when tasks are delegated.

Delegator:

In healthcare, a delegator is an individual, typically a healthcare professional or provider, who assigns specific tasks, responsibilities, or decision-making authority to another person or entity, known as the delegatee. Delegation is a critical aspect of effective healthcare delivery, as it allows for the distribution of responsibilities among team members to optimize workflow and enhance patient care. Delegators must carefully assess the competence, qualifications, and training of potential delegatees, ensuring that they possess the requisite skills to carry out delegated tasks safely and effectively. While delegation is a practical means to streamline healthcare processes, delegators bear the ultimate responsibility for the outcomes of the delegated activities, maintaining accountability for patient well-being, compliance with regulations, and the overall quality of care provided by the healthcare team. Clear communication, trust, and a thorough understanding of legal and ethical considerations are integral to successful delegation in the healthcare domain.

PKG:

The term "pkg" can refer to various things depending on the context. In software development, particularly in macOS systems, "pkg" often stands for a package, which is a format for software distribution and installation. A package file typically contains compressed application files, scripts, and metadata necessary for the installation process. These packages simplify the deployment of software by bundling everything needed for installation into a single file, easing the distribution and installation processes for end-users. Additionally, "pkg" may also refer to "package" in a more general sense, representing a bundled set of files or resources designed for a specific purpose, such as a software library or a collection of related components. The interpretation of "pkg" depends on the specific domain, and additional context is needed for a more precise understanding.

CLOUD SERVER:

At its core, the module will employ robust encryption techniques to protect data both in transit and at rest. Utilizing state-of-the-art cryptographic algorithms, such as AES (Advanced Encryption Standard) and RSA (Rivest-Shamir-Adleman), data will be encrypted prior to transmission and securely stored within the cloud server.

9. SOFTWARE ENVIRONMENT

9.1 JAVA TECHNOLOGY

Java technology is both a programming language and a platform Java technology is a versatile and widely adopted programming language known for its platform independence, object-oriented design, and robust features. Introduced by Sun Microsystems and now maintained by Oracle, Java follows the "Write Once, Run Anywhere" philosophy, enabling developers to create applications that can run on diverse platforms with a Java Virtual Machine (JVM). With features like automatic memory management, multithreading support, and a rich standard library, Java is utilized across a spectrum of applications, from web development (JavaServer Pages, Servlets) and mobile apps (Android) to enterprise-levelsystems (Java EE). Its active community, extensive ecosystem, and ongoing updates contribute to its enduring popularity in the software development landscape.

Java technology does not require an introduction. Everyone around the world is still amazedat the astonishing power of Java in web and mobile development. Of course, you too may be tempted by Java's popularity and monopoly in software development, and you may want to use Java programming language with your next web development solution.

Java lets you process complex applications' solutions like tally voting polls, flight booking APIs, hotel booking, reservation systems, and more. However, you would not know whatJava technologies you need to develop a complex or simple web application?

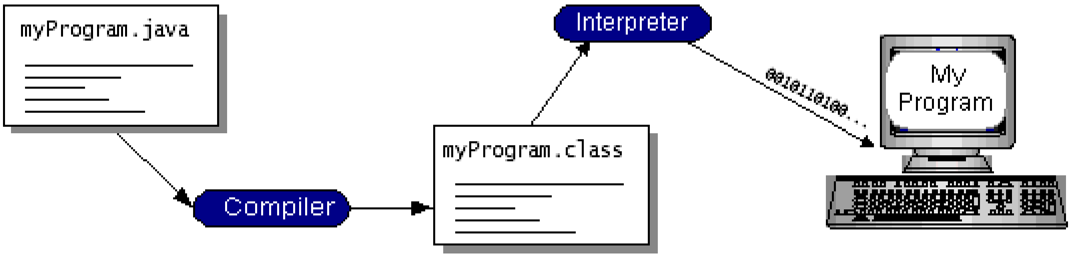
There is a wide range of technologies under the term Java that describe web development in various ways. In this section, we have discussed about Java technologies that can used to build a web application. Whether you are a full-stack developer, a back-end or front-end developer, or a business owner, knowledge of these technologies is essential to understandingthe flow of your application.

The Java Programming Language

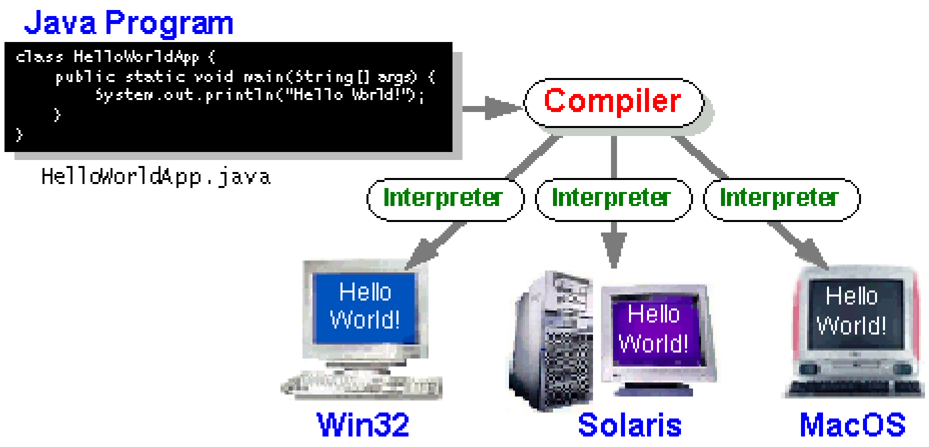
The Java programming language is a high-level language that can be characterized by all of the following buzzwords:

1. Simple
2. Architecture neutral
3. Object oriented
4. Portable
5. Distributed
6. High performance
7. Interpreted
8. Multithreaded
9. Robust
10. Dynamic
11. Secure

With most programming languages, you either compile or interpret a program so that you can run it on your computer. The Java programming language is unusual in that a program is both compiled and interpreted. With the compiler, first you translate a program into an intermediate language called Java byte codes —the platform-independent codes interpreted by the interpreter on the Java platform. The interpreter parses and runs each Java byte code instruction on the computer. Compilation happens just once; interpretation occurs each time the program is executed. The following figure illustrates how this works.



You can think of Java byte codes as the machine code instructions for the Java Virtual Machine (Java VM). Every Java interpreter, whether it's a development tool or a Web browser that can run applets, is an implementation of the Java VM. Java byte codes help make “write once, run anywhere” possible. You can compile your program into byte codes on any platform that has a Java compiler. The byte codes can then be run on any implementation of the Java VM. That means that as long as a computer has a Java VM, the same program written in the Java programming language can run on Windows 2000, a Solaris workstation, or on an iMac.



```
class HelloWorldApp {
    public static void main(String[] args) {
        System.out.println("Hello World!");
    }
}
```

The Java Platform

A platform is the hardware or software environment in which a program runs. We’ve already mentioned some of the most popular platforms like Windows 2000, Linux, Solaris, and MacOS. Most platforms can be described as a combination of the operating system and hardware. The Java platform differs from most other platforms in that it’s a software-only platform that runs on top of other hardware-based platforms.

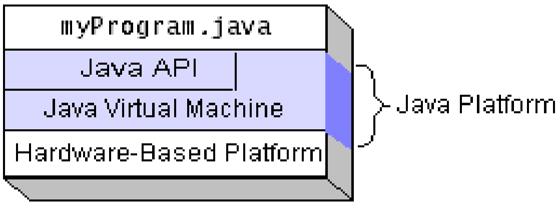
The Java platform has two components:

1. The *Java Virtual Machine* (Java VM)
2. The *Java Application Programming Interface* (Java API)

You’ve already been introduced to the Java VM. It’s the base for the Java platform and is ported onto various hardware-based platforms.

The Java API is a large collection of ready-made software components that provide many useful capabilities, such as graphical user interface (GUI) widgets. The Java API is grouped into libraries of related classes and interfaces; these libraries are known as packages. The next section, What Can Java Technology Do? Highlights what functionality some of the packages in the Java API provide.

The following figure depicts a program that's running on the Java platform. As the figure shows, the Java API and the virtual machine insulate the program.



Native code is code that after you compile it, the compiled code runs on a specific hardware platform. As a platform-independent environment, the Java platform can be a bit slower than native code. However, smart compilers, well-tuned interpreters, and just-in-time byte code compilers can bring performance close to that of native code without threatening portability. What Can Java Technology Do?

The most common types of programs written in the Java programming language are applets and applications. If you have surfed the Web, you're probably already familiar with applets. An applet is a program that adheres to certain conventions that allow it to run within a Java- enabled browser.

However, the Java programming language is not just for writing cute, entertaining applets for the Web. The general-purpose, high-level Java programming language is also a powerful software platform. Using the generous API, you can write many types of programs.

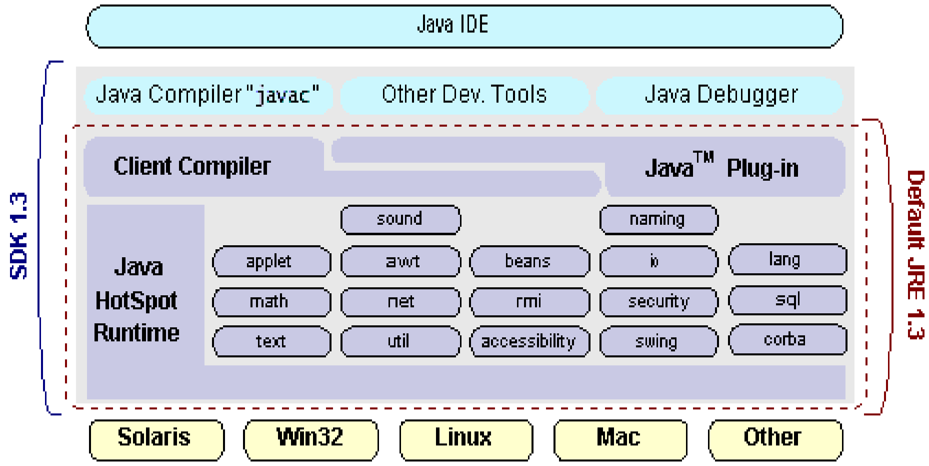
An application is a standalone program that runs directly on the Java platform. A special kind of application known as a server serves and supports clients on a network. Examples servers are Web servers, proxy servers, mail servers, and print servers. Another specialized program is a servlet. A servlet can almost be thought of as an applet that runs on the server side. Java Servlets are a popular choice for building interactive web applications, replacing the use of CGI scripts.

Servlets are similar to applets in that they are runtime extensions of applications. Instead of working in browsers, though, servlets run within Java Web servers, configuring or tailoring the server.

How does the API support all these kinds of programs? It does so with packages of software components that provides a wide range of functionality. Every full implementation of the Java platform gives you the following features:

1. **The essentials**: Objects, strings, threads, numbers, input and output, data structures, system properties, date and time, and so on.
2. **Applets**: The set of conventions used by applets.
3. **Networking**: URLs, TCP (Transmission Control Protocol), UDP (User Data gram Protocol) sockets, and IP (Internet Protocol) addresses.
4. **Internationalization**: Help for writing programs that can be localized for users worldwide. Programs can automatically adapt to specific locales and be displayed in the appropriate language.
5. **Security**: Both low level and high level, including electronic signatures, public and private key management, access control, and certificates.
6. **Software components**: Known as JavaBeans^{TM}, can plug into existing component architectures.
7. **Object serialization**: Allows lightweight persistence and communication via Remote Method Invocation (RMI).
8. **Java Database Connectivity (JDBC^{TM})**: Provides uniform access to a wide range of relational databases.

The Java platform also has APIs for 2D and 3D graphics, accessibility, servers, collaboration, telephony, speech, animation, and more. The following figure depicts what is included in the Java 2 SDK.



Java Database Connectivity

What Is JDBC?

JDBC is a Java API for executing SQL statements. (As a point of interest, JDBC is a trademarked name and is not an acronym; nevertheless, JDBC is often thought of as standing for Java Database Connectivity. It consists of a set of classes and interfaces written in the Java programming language. JDBC provides a standard API for tool/database developers and makes it possible to write database applications using a pure Java API.

Using JDBC, it is easy to send SQL statements to virtually any relational database. One can write a single program using the JDBC API, and the program will be able to send SQL statements to the appropriate database. The combinations of Java and JDBC lets a programmer write it once and run it anywhere.

Simply put, JDBC makes it possible to do three things:

- Establish a connection with a database
- Send SQL statements
- Process the results.

JDBC versus ODBC and other APIs

At this point, Microsoft's ODBC (Open Database Connectivity) API is that probably the most widely used programming interface for accessing relational databases. It offers the ability to connect to almost all databases on almost all platforms.

So why not just use ODBC from Java? The answer is that you can use ODBC from Java, but this is best done with the help of JDBC in the form of the JDBC-ODBC Bridge, which we will cover shortly. The question now becomes "Why do you need JDBC?" There are several answers to this question:

1. ODBC is not appropriate for direct use from Java because it uses a C interface. Calls from Java to native C code have a number of drawbacks in the security, implementation, robustness, and automatic portability of applications.
2. A literal translation of the ODBC C API into a Java API would not be desirable. For

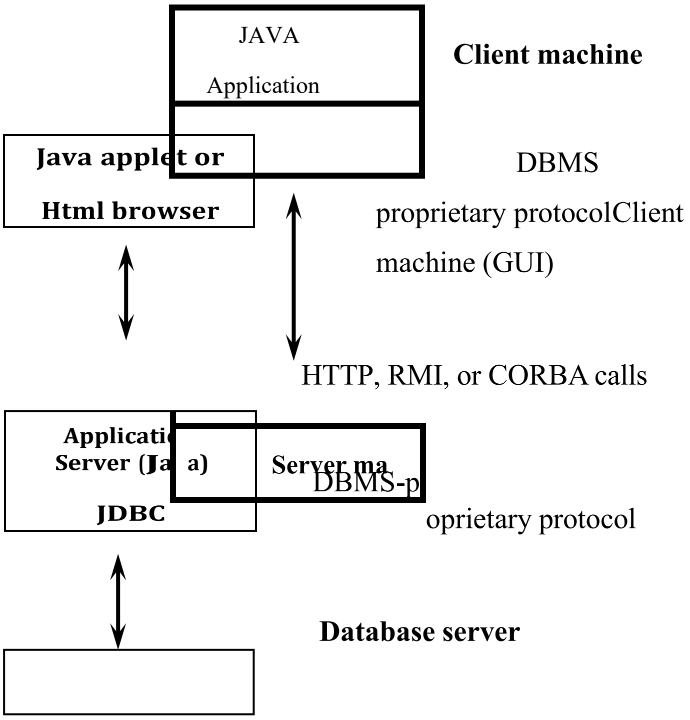
example, Java has no pointers, and ODBC makes copious use of them, including the notoriously error-prone generic pointer "void *". You can think of JDBC as ODBC translated into an object-oriented interface that is natural for Java programmers.

3. ODBC is hard to learn. It mixes simple and advanced features together, and it has complex options even for simple queries. JDBC, on the other hand, was designed to keep simple things simple while allowing more advanced capabilities where required.
4. A Java API like JDBC is needed in order to enable a "pure Java" solution. When ODBC is used, the ODBC driver manager and drivers must be manually installed on every client machine. When the JDBC driver is written completely in Java, however, JDBC code is automatically installable, portable, and secure on all Java platforms fromnetwork computers to mainframes.

Two-tier and Three-tier Models

In the two-tier model, a Java applet or application talks directly to the database. This requires a JDBC driver that can communicate with the particular database management system being accessed. A user's SQL statements are delivered to the database, and the results of those statements are sent back to the user. The database may be located on another machine to which the user is connected via a network. This is referred to as a client/server configuration, with the user's machine as the client, and the machine housing the database as the server. The network can be an Intranet, which, for example, connects employees within a corporation, or it can be the Internet.

In the three-tier model, commands are sent to a "middle tier" of services, which then send SQL statements to the database. The database processes the SQL statements and sends the results back to the middle tier, which then sends them to the user. MIS directors find the three-tier model very attractive because the middle tier makes it possible to maintain control over access and the kinds of updates that can be made to corporate data. Another advantage is that when there is a middle tier, the user can employ an easy-to-use higher-level API which is translated by the middle tier into the appropriate low-level calls. Finally, in many cases the three-tier architecture can provide performance advantages.



Until now the middle tier has typically been written in languages such as C or C++, which offer fast performance. However, with the introduction of optimizing compilers that translate Java byte code into efficient machine-specific code, it is becoming practical to implement the middle tier in Java. This is a big plus, making it possible to take advantage of Java's robustness, multithreading, and security features. JDBC is important to allow database access from a Java middle tier.

JDBC Driver Types

The JDBC drivers that we are aware of at this time fit into one of four categories:

- JDBC-ODBC bridge plus ODBC driver
- Native-API partly-Java driver
- JDBC-Net pure Java driver
- Native-protocol pure Java driver

JDBC-ODBC Bridge

If possible, use a Pure Java JDBC driver instead of the Bridge and an ODBC driver. This completely eliminates the client configuration required by ODBC. It also eliminates the potential that the Java VM could be corrupted by an error in the native code brought in by the Bridge (that is, the Bridge native library, the ODBC driver manager library, the ODBC driver library, and the database client library).

What Is the JDBC- ODBC Bridge?

The JDBC-ODBC Bridge is a JDBC driver, which implements JDBC operations by translating them into ODBC operations. To ODBC it appears as a normal application program. The Bridge implements JDBC for any database for which an ODBC driver is available. The Bridge is implemented as the sun.jdbc.odbc Java package and contains a native library used to access ODBC. The Bridge is a joint development of Intersolv and JavaSoft.

Java Server Pages (JSP)

Java server Pages is a simple, yet powerful technology for creating and maintaining dynamic- content web pages. Based on the Java programming language, Java Server Pages offers proven portability, open standards, and a mature re-usable component model .The Java Server Pages architecture enables the separation of content generation from content presentation. This separation not eases maintenance headaches, it also allows web team members to focus on their areas of expertise. Now, web page designer can concentrate on layout, and web application designers on programming, with minimal concern about impacting each other's work.

Features of JSP

Portability:

Java Server Pages files can be run on any web server or web-enabled application server that provides support for them. Dubbed the JSP engine, this support involves recognition, translation, and management of the Java Server Page lifecycle and its interaction components.

Components

It was mentioned earlier that the Java Server Pages architecture can include reusable Java components. The architecture also allows for the embedding of a scripting language directly into the Java Server Pages file. The components current supported include Java Beans, and Servlets.

Processing

A Java Server Pages file is essentially an HTML document with JSP scripting or tags. The Java Server Pages file has a JSP extension to the server as a Java Server Pages file. Before the page is served, the Java Server Pages syntax is parsed and processed into a Servlet on the server side. The Servlet that is generated outputs real content in straight HTML for responding to the client.

Access Models:

A Java Server Pages file may be accessed in at least two different ways. A client's request comes directly into a Java Server Page. In this scenario, suppose the page accesses reusable Java Bean components that perform particular well-defined computations like accessing a database. The result of the Beans computations, called result sets is stored within the Bean as properties. The page uses such Beans to generate dynamic content and present it back to the client.

In both of the above cases, the page could also contain any valid Java code. Java Server Pages architecture encourages separation of content from presentation.

Steps in the execution of a JSP Application:

1. The client sends a request to the web server for a JSP file by giving the name of the JSP file within the form tag of a HTML page.
2. This request is transferred to the JavaWebServer. At the server side JavaWebServer receives the request and if it is a request for a jsp file server gives this request to the JSP engine.
3. JSP engine is program which can understands the tags of the jsp and then it converts those tags into a Servlet program and it is stored at the server side. This Servlet is loaded in the memory and then it is executed and the result is given back to the JavaWebServer and then it is transferred back to the result is given back to the JavaWebServer and then it is transferred back to the client.

JDBC connectivity

The JDBC provides database-independent connectivity between the J2EE platform and a wide range of tabular data sources. JDBC technology allows an Application Component Provider to:

- Perform connection and authentication to a database server
- Manager transactions

- Move SQL statements to a database engine for preprocessing and execution
- Execute stored procedures
- Inspect and modify the results from Select statements.

Tomcat 6.0 web server

Tomcat is an open source web server developed by Apache Group. Apache Tomcat is the servlet container that is used in the official Reference Implementation for the Java Servlet andJava Server Pages technologies. The Java Servlet and Java Server Pages specifications are developed by Sun under the Java Community Process. Web Servers like Apache Tomcat support only web components while an application server supports web components as wellas business components (BEAs Weblogic, is one of the popular application server).To develop web application with jsp/servlet install any web server like JRun, Tomcat your application.

How Will Java Technology Change My Life?

We can't promise you fame, fortune, or even a job if you learn the Java programming language. Still, it is likely to make your programs better and requires less effort than other languages. We believe that Java technology will help you do the following:

1. **Get started quickly**: Although the Java programming language is a powerful object- oriented language, it's easy to learn, especially for programmers already familiar with C or C++.
2. **Write less code**: Comparisons of program metrics (class counts, method counts, and so on) suggest that a program written in the Java programming language can be four times smaller than the same program in C++.
3. **Write better code**: The Java programming language encourages good coding practices, and its garbage collection helps you avoid memory leaks. Its object orientation, its JavaBeans component architecture, and its wide-ranging, easily extendible API let you reuse other people's tested code and introduce fewer bugs.
4. **Develop programs more quickly**: Your development time may be as much as twice as fast versus writing the same program in C++. Why? You write fewer lines of code and it is a simpler programming language than C++.
5. **Avoid platform dependencies with 100% Pure Java**: You can keep your program portable by avoiding the use of libraries written in other languages. The 100% Pure

Java^{TM} Product Certification Program has a repository of historical process manuals, white papers, brochures, and similar materials online.

6. **Write once, run anywhere**: Because 100% Pure Java programs are compiled into machine-independent byte codes, they run consistently on any Java platform.
7. **Distribute software more easily**: You can upgrade applets easily from a central server. Applets take advantage of the feature of allowing new classes to be loaded “onthe fly,” without recompiling the entire program.

ODBC

Microsoft Open Database Connectivity (ODBC) is a standard programming interface for application developers and database systems providers. Before ODBC became a *de facto* standard for Windows programs to interface with database systems, programmers had to use proprietary languages for each database they wanted to connect to. Now, ODBC has made thechoice of the database system almost irrelevant from a coding perspective, which is as it should be. Application developers have much more important things to worry about than the syntax that is needed to port their program from one database to another when business needssuddenly change.

Through the ODBC Administrator in Control Panel, you can specify the particular database that is associated with a data source that an ODBC application program is written to use. Think of an ODBC data source as a door with a name on it. Each door will lead you to a particular database. For example, the data source named Sales Figures might be a SQL Server database, whereas the Accounts Payable data source could refer to an Access database. The physical database referred to by a data source can reside anywhere on the LAN.

The ODBC system files are not installed on your system by Windows 95. Rather, they are installed when you setup a separate database application, such as SQL Server Client or Visual Basic 4.0. When the ODBC icon is installed in Control Panel, it uses a file called ODBCINST.DLL. It is also possible to administer your ODBC data sources through a stand- alone program called ODBCADM.EXE. There is a 16-bit and a 32-bit version of this program and each maintains a separate list of ODBC data sources.

From a programming perspective, the beauty of ODBC is that the application can be written to use the same set of function calls to interface with any data source, regardless of the database vendor. The source code of the application doesn’t change whether it talks to Oracle or SQL Server. We only mention these two as an example. There are ODBC drivers available for several dozen popular database systems. Even Excel spreadsheets and plain text files can be turned into data sources. The operating system uses the Registry information written by ODBC Administrator to determine which low-level ODBC drivers are needed to talk to the data source (such as the interface to Oracle or SQL Server). The loading of the ODBC drivers is transparent to the ODBC application program. In a client/server environment, the ODBC API even handles many of the network issues for the application programmer.

The advantages of this scheme are so numerous that you are probably thinking there must be some catch. The only disadvantage of ODBC is that it isn’t as efficient as talking directly to the native database interface. ODBC has had many detractors make the charge that it is too slow. Microsoft has always claimed that the critical factor in performance is the quality of the driver software that is used. In our humble opinion, this is true. The availability of good ODBC drivers has improved a great deal recently. And anyway, the criticism about performance is somewhat analogous to those who said that compilers would never match the speed of pure assembly language. Maybe not, but the compiler (or ODBC) gives you the opportunity to write cleaner programs, which means you finish sooner. Meanwhile, computers get faster every year.

JDBC

In an effort to set an independent database standard API for Java; Sun Microsystems developed Java Database Connectivity, or JDBC. JDBC offers a generic SQL database access mechanism that provides a consistent interface to a variety of RDBMSs. This consistent interface is achieved through the use of “plug-in” database connectivity modules, or drivers. If a database vendor wishes to have JDBC support, he or she must provide the driver for each platform that the database and Java run on.

To gain a wider acceptance of JDBC, Sun based JDBC’s framework on ODBC. As you discovered earlier in this chapter, ODBC has widespread support on a variety of platforms. Basing JDBC on ODBC will allow vendors to bring JDBC drivers to market much faster than developing a completely new connectivity solution.

JDBC was announced in March of 1996. It was released for a 90 day public review that ended June 8, 1996. Because of user input, the final JDBC v1.0 specification was released soon after.

The remainder of this section will cover enough information about JDBC for you to know what it is about and how to use it effectively. This is by no means a complete overview of JDBC. That would fill an entire book.

JDBC Goals

Few software packages are designed without goals in mind. JDBC is one that, because of its many goals, drove the development of the API. These goals, in conjunction with early reviewer feedback, have finalized the JDBC class library into a solid framework for building database applications in Java.

The goals that were set for JDBC are important. They will give you some insight as to why certain classes and functionalities behave the way they do. The eight design goals for JDBC are as follows:

1. SQL Level API

The designers felt that their main goal was to define a SQL interface for Java. Although not the lowest database interface level possible, it is at a low enough level for higher-level tools and APIs to be created. Conversely, it is at a high enough level for application programmers to use it confidently. Attaining this goal allows for future tool vendors to “generate” JDBC code and to hide many of JDBC’s complexities from the end user.

2. SQL Conformance

SQL syntax varies as you move from database vendor to database vendor. In an effort to support a wide variety of vendors, JDBC will allow any query statement to be passed through it to the underlying database driver. This allows the connectivity module to handle non-standard functionality in a manner that is suitable for its users.

3.JDBC must be implemental on top of common database interfaces

The JDBC SQL API must “sit” on top of other common SQL level APIs. This goal allows JDBC to use existing ODBC level drivers by the use of a software interface. This interface would translate JDBC calls to ODBC and vice versa.

4.Provide a Java interface that is consistent with the rest of the Java

system

Because of Java’s acceptance in the user community thus far, the designers feel that they should not stray from the current design of the core Java system.

4.Keep it simple

This goal probably appears in all software design goal listings. JDBC is no exception. Sun felt that the design of JDBC should be very simple, allowing for only one method of completing a task per mechanism. Allowing duplicate functionality only serves to confuse the users of the API.

3. Use strong, static typing wherever possible

Strong typing allows for more error checking to be done at compile time; also, less error appear at runtime.

4. Keep the common cases simple

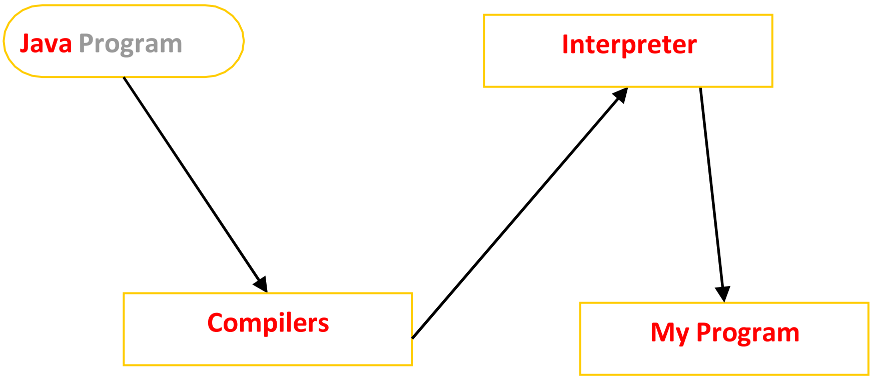
Because more often than not, the usual SQL calls used by the programmer are simple SELECT’s, INSERT’s, DELETE’s and UPDATE’s, these queries should be simple to perform with JDBC. However, more complex SQL statements should also be possible.

Finally we decided to proceed the implementation using Java Networking and for dynamically updating the cache table we go for MS Access database.

Java ha two things: a programming language and a platform Java is a high-level programming language that is all of the following

Simple
Object-oriented
Distributed
Interpreted
Robust
Architecture-neutral
Portable
High-performance
multithreaded

Java is also unusual in that each Java program is both compiled and interpreted. With a compile you translate a Java program into an intermediate language called Java byte codes the platform-independent code instruction is passed and run on the computer.Compilation happens just once; interpretation occurs each time the program is executed. The figure illustrates how this works.

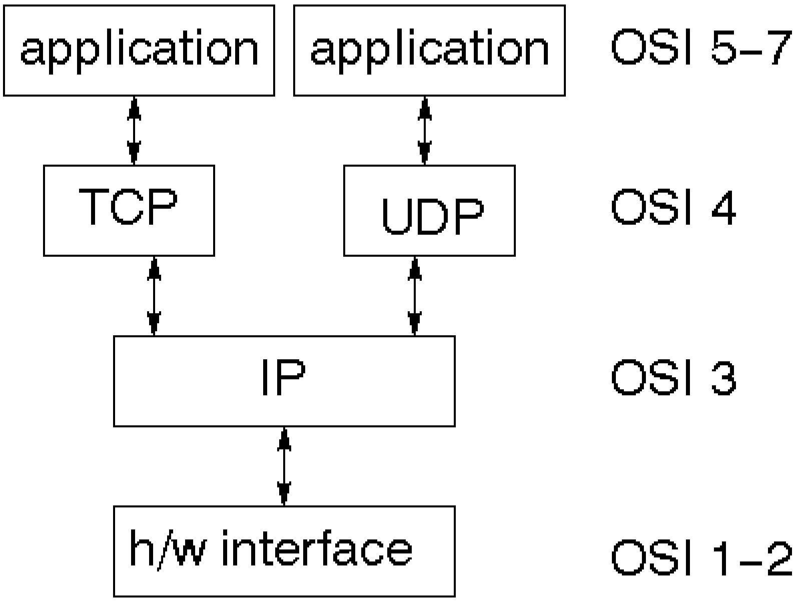


You can think of Java byte codes as the machine code instructions for the Java Virtual Machine (Java VM). Every Java interpreter, whether it's a Java development tool or a Web browser that can run Java applets, is an implementation of the Java VM. The Java VM can also be implemented in hardware.

Java byte codes help make "write once, run anywhere" possible. You can compile your Java program into byte codes on my platform that has a Java compiler.

Networking TCP/IP stack

The TCP/IP stack is shorter than the OSI one:



TCP is a connection-oriented protocol; UDP (User Datagram Protocol) is a connectionless protocol.

IP datagram's

The IP layer provides a connectionless and unreliable delivery system. It considers each datagram independently of the others. Any association between datagram must be supplied by the higher layers. The IP layer supplies a checksum that includes its own header. The header includes the source and destination addresses. The IP layer handles routing through an Internet. It is also responsible for breaking up large datagram into smaller ones for transmission and reassembling them at the other end.

UDP

UDP is also connectionless and unreliable. What it adds to IP is a checksum for the contents of the datagram and port numbers. These are used to give a client/server model - see later.

TCP

TCP supplies logic to give a reliable connection-oriented protocol above IP. It provides a virtual circuit that two processes can use to communicate.

Internet addresses

Inorder to use a service, you must be able to find it. The Internet uses an address scheme for machines so that they can be located. The address is a 32 bit integer which gives the IP address. This encodes a network ID and more addressing. The network ID falls into various classes according to the size of the network address.

Network address

Class A uses 8 bits for the network address with 24 bits left over for other addressing. Class B uses 16 bit network addressing. Class C uses 24 bit network addressing and class D uses all 32.

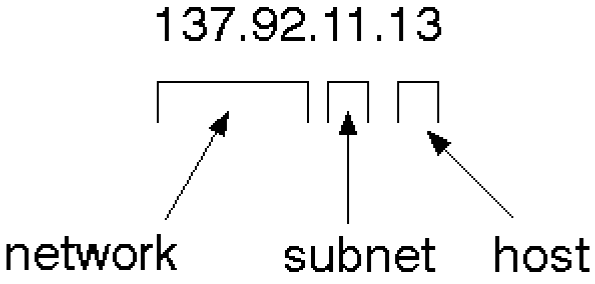
Subnet address

Internally, the UNIX network is divided into sub networks. Building 11 is currently on one sub network and uses 10-bit addressing, allowing 1024 different hosts.

Host address

8 bits are finally used for host addresses within our subnet. This places a limit of 256 machines that can be on the subnet.

Total address



The 32 bit address is usually written as 4 integers separated by dots.

Port addresses

A service exists on a host, and is identified by its port. This is a 16 bit number. To send a message to a server, you send it to the port for that service of the host that it is running on. This is not location transparency! Certain of these ports are "well known".

Sockets

A socket is a data structure maintained by the system to handle network connections. A socket is created using the call socket. It returns an integer that is like a file descriptor. In fact, under Windows, this handle can be used with Read File and Write File functions.

```
#include <sys/types.h> #include <sys/socket.h>
```

```
int socket(int family, int type, int protocol);
```

Here "family" will be AF_INET for IP communications, protocol will be zero, and type will depend on whether TCP or UDP is used. Two processes wishing to communicate over a network create a socket each. These are similar to two ends of a pipe - but the actual pipe does not yet exist.

JFree Chart

JFreeChart is a free 100% Java chart library that makes it easy for developers to display professional quality charts in their applications. JFreeChart's extensive feature set includes:

A consistent and well-documented API, supporting a wide range of chart types;

A flexible design that is easy to extend, and targets both server-side and client-side applications;

Support for many output types, including Swing components, image files (including PNG and JPEG), and vector graphics file formats (including PDF, EPS and SVG);

JFreeChart is "open source" or, more specifically, free software. It is distributed under the terms of the GNU Lesser General Public Licence (LGPL), which permits use in proprietary applications.

1. Map Visualizations

Charts showing values that relate to geographical areas. Some examples include: (a) population density in each state of the United States, (b) income per capita for each country in Europe, (c) life expectancy in each country of the world. The tasks in this project include:

1.Sourcing freely redistributable vector outlines for the countries of the world, states/provinces in particular countries (USA in particular, but also other areas);

2.Creating an appropriate dataset interface (plus default implementation), a rendered, and integrating this with the existing XY Plot class in JFree Chart;

3.Testing, documenting, testing some more, documenting some more.

2.Time Series Chart Interactivity:

Implement a new (to JFreeChart) feature for interactive time series charts --- to display a separate control that shows a small version of ALL the time series data, with a sliding "view" rectangle that allows you to select the subset of the time series data to display in the main chart.

3.Dashboards

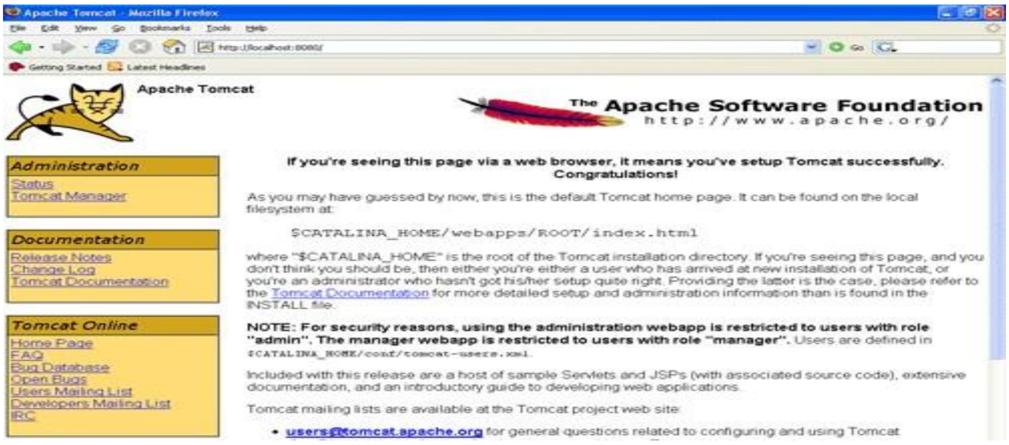
There is currently a lot of interest in dashboard displays. Create a flexible dashboard mechanism that supports a subset of JFreeChart chart types (dials, pies, thermometers, bars, and lines/time series) that can be delivered easily via both Java Web Start and an applet.

4. Property Editors

The property editor mechanism in JFreeChart only handles a small subset of the properties that can be set for charts. Extend (or reimplement) this mechanism to provide greater end- user control over the appearance of the charts.

Tomcat 6.0 web server

Tomcat is an open source web server developed by Apache Group. Apache Tomcat is the servlet container that is used in the official Reference Implementation for the Java Servlet and JavaServer Pages technologies. The Java Servlet and Java Server Pages specifications are developed by Sun under the Java Community Process. Web Servers like Apache Tomcat support only web components while an application server supports web components as well as business components (BEAs Web logic, is one of the popular application server).To develop a web application with jsp/servlet install any web server like JRun, Tomcat etc to run your application.



If you're seeing this page via a web browser, it means you've setup Tomcat successfully. Congratulations!

As you may have guessed by now, this is the default Tomcat home page. It can be found on the local filesystem at:

```
$CATALINA_HOME/webapps/ROOT/index.html
```

where "$CATALINA_HOME" is the root of the Tomcat installation directory. If you're seeing this page, and you don't think you should be, then either you're either a user who has arrived at new installation of Tomcat, or you're an administrator who hasn't got his/her setup quite right. Providing the latter is the case, please refer to the Tomcat Documentation for more detailed setup and administration information than is found in the INSTALL file.

NOTE: For security reasons, using the administration webapp is restricted to users with role "admin". The manager webapp is restricted to users with role "manager". Users are defined in `$CATALINA_HOME/conf/tomcat-users.xml`.

Included with this release are a host of sample Servlets and JSPs (with associated source code), extensive documentation, and an introductory guide to developing web applications.

Tomcat mailing lists are available at the Tomcat project web site:

- users@tomcat.apache.org for general questions related to configuring and using Tomcat

Fig Tomcat Webserver

What is a J2ME profile?

As we mentioned earlier in this tutorial, a profile defines the type of device supported. The Mobile Information Device Profile (MIDP), for example, defines classes for cellular phones. It adds domain-specific classes to the J2ME configuration to define uses for similar devices. Two profiles have been defined for J2ME and are built upon CLDC: KJava and MIDP. Both KJava and MIDP are associated with CLDC and smaller devices. Profiles are built on top of configurations. Because profiles are specific to the size of the device (amount of memory) on which an application runs, certain profiles are associated with certain configurations.

A skeleton profile upon which you can create your own profile, the Foundation Profile, is available for CDC.

Profile 1: KJava

KJava is Sun's proprietary profile and contains the KJava API. The KJava profile is built on top of the CLDC configuration. The KJava virtual machine, KVM, accepts the same byte codes and class file format as the classic J2SE virtual machine. KJava contains a Sun-specific API that runs on the Palm OS. The KJava API has a great deal in common with the J2SE Abstract Windowing Toolkit (AWT). However, because it is not a standard J2ME package, its main package is com.sun.kjava. We'll learn more about the KJava API later in this tutorial when we develop some sample applications.

Profile 2: MIDP

MIDP is geared toward mobile devices such as cellular phones and pagers. The MIDP, like KJava, is built upon CLDC and provides a standard run-time environment that allows new applications and services to be deployed dynamically on end user devices. MIDP is a common, industry-standard profile for mobile devices that is not dependent on a specific vendor. It is a complete and supported foundation for mobile application

development. MIDP contains the following packages, the first three of which are core CLDC packages, plus three MIDP-specific packages.

* java.io
* java.lang
* java.util
* javax.microedition.io
* javax.microedition.lcdui
* javax. microedition.midlet
* javax.microedition.rms

9.2 SOURCE CODE

index.html

```
<html>

<head>

<title>Secure Keyword Search</title>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />

<link rel="stylesheet" href="layout/styles/layout.css" type="text/css" />

<script type="text/javascript" src="layout/scripts/jquery.min.js"></script>

<scripttype="text/javascript" src="layout/scripts/jquery.innerfade.js"></script>

</head>

<body id="top">

<div class="wrapper col1">
```

```
<div id="header">
<center>
<h2>Secure Keyword Search and Data Sharing Mechanism</h2>
</center>
<div id="topnav" style="width:auto;">
<ul>
<li class="last"><a href="PKG_Login.jsp">PKG</a></li>
<!-- <li><a href="CloudServer.jsp">Cloud Server</a></li>-->
<li><a href="Delegatee.jsp">Delegatee</a></li>
<li><a href="Delegator.jsp">Delegator</a></li>
<li><a href="HealthRecordOwner.jsp">Health Record Owner</a></li>
<li class="active"><a href="index.html">Home</a></li>
</ul>
</div>
<br class="clear" />
</div>
</div>
<!--
####################################################################
######################################### -->
<div class="wrapper col2">
<div id="intro">
<div class="fl_left">
<h1>About the Project</h1>
<p align="justify">
```

The emergence of cloud infrastructure has significantly reduced the costs of hardware and software resources in computing infrastructure. To ensure security, the data is usually encrypted before it's outsourced to the cloud. Unlike searching and sharing the plain data, it is challenging to search and share the data after encryption. Nevertheless, it is a critical task for the cloud service provider as the users expect the cloud to conduct a quick search and return the result without losing data confidentiality. To overcome these problems, we propose a ciphertext-policy attribute-based mechanism with keyword search and data sharing (CPAB- KSDS) for encrypted cloud data. The proposed solution not only supports attribute-based keyword search but also enables attribute-based data sharing at the same time, which is in contrast to the existing solutions that only support either one of two features. Additionally,the keyword in our scheme can be updated during the sharing phase without interacting with the PKG. In this paper, we describe the notion of CPAB-KSDS as well as its security model. Besides, we propose a concrete scheme and prove that it is against chosen ciphertext attack and chosen keyword attack secure in the random oracle model. Finally, the proposed construction is demonstrated practical and efficient in the performance and property comparison.

```
        </p>

        </div>

        <div class="fl_right">

        <ul id="rotation">

        <li><a href="#"><img src="images/Capture.PNG" alt="" /></a></li>

        <li><a href="#"><img src="images/img.png" alt="" /></a></li>

        <li><a href="#"><img src="images/img2.jpg" alt="" /></a></li>

        </ul>

        </div>

        <br class="clear" />

        </div>

        </div>

        <!--
################################################################
####################################### -->
```

```
<div class="wrapper col3">
<div id="container">

<div id="latest">



<br class="clear" />

</div>

<div class="clear"></div>

</div>

</div>

<!--
#########################################################################
############################################# -->
<div class="wrapper col5">
<div id="copyright">

<p class="fl_left">Secure Keyword Search and Data Sharing Mechanism</p>

<p class="fl_right"></p>

<br class="clear" />

</div>

</div>

</body>

</html>
```

10. RESULTS/DISCUSSION

10.1 SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

TYPES OF TESTS:

Unit testing

Functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Functional testing

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is cantered on the following items:

Valid Input : identified classes of valid input must be accepted. Invalid Input : identified classes of invalid input must be rejected. Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised. Systems/Procedures : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

SYSTEM TESTING

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software work

unit Testing:

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach:

Field testing will be performed manually and functional tests will be written in detail.

Test objectives:

1. 1.All field entries must work properly.
2. Pages must be activated from the identified link.
3. The entry screen, messages and responses must not be delayed.

Features to be tested:

1.Verify that the entries are of the correct format

2.No duplicate entries should be allowed

3.All links should take the user to the correct page.

Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects. The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

Acceptance Testing: User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

10.1.1 TEST CASES

Test case1:

Test case for health record owner registration form

FUNCTION:	registration
EXPECTED RESULTS:	Should check if all the fields are filled by the client and saving the client to database
ACTUAL RESULTS:	Checking whether all the fields are field by client or not through validations and saving user.
LOW PRIORITY	No
HIGH PRIOR ITY	Yes

Test case2:

Test case for health record owner login form

FUNCTION:	Owner LOGIN
EXPECTED RESULTS:	Should check if all the fields are filled by the client and saving the client to database
ACTUAL RESULTS:	Checking whether all the fields are field by client or not through validations and saving user.
LOW PRIORITY	NO

HIGH PRIORITY	YES

Test case3:

Test case for delegatee registration form:

FUNCTION:	DELEGATEE REGISTRATION
EXPECTED RESULTS:	Should check if all the fields are filled by the client and saving the client to database.
ACTUAL RESULTS:	Checking whether all the fields are field by client or not through validations and saving user.
LOW PRIORITY	NO
HIGH PRIORITY	YES

Test case4:

Test case for delegatee login:

When the client used to login he/she can enter username ,password to view the profile ,to upload documents, edit and delete documents and etc , then this results in displaying an client menu.

FUNCTION:	delegatee LOGIN
EXPECTED RESULTS:	Should check if all the fields are filled by the client and saving the client to database
ACTUAL RESULTS:	Checking whether all the fields are field by client or not through validations and saving user.

LOW PRIORITY	NO
HIGH PRIORITY	YES

Test case 5:

Test case for delegator registration form:

FUNCTION:	DELEGATOR REGISTRATION
EXPECTED RESULTS:	Should check if all the fields are filled by the client and saving the client to database.
ACTUAL RESULTS:	Checking whether all the fields are field by client or not through validations and saving user.
LOW PRIORITY	NO
HIGH PRIORITY	YES

Test case 6:

Test case for delegator login**:**

When the client used to login he/she can enter username,password to view the profile ,to upload documents, edit and delete documents and etc , then this results in displaying an client menu.

FUNCTION:	delegator LOGIN
EXPECTED RESULTS:	Should check if all the fields are filled by the client and saving the client to database
ACTUAL RESULTS:	Checking whether all the fields are field by client or not through validations and saving user.

LOW PRIORITY	NO
HIGH PRIORITY	YES

Test case 7: Test case for cloud login:

When the cloud used to login he/she can enter username password to view all users p to view all documents, view all transactions and etc , then this results in displaying an cloud menu.

FUNCTION:	CLOUD LOGIN
EXPECTED RESULTS:	Should check if all the fields are filled by the client and saving the client to database
ACTUAL RESULTS:	Checking whether all the fields are field by client or not through validations and saving user.
LO PRIORITY	NO
HIGH PRIORITY	YES

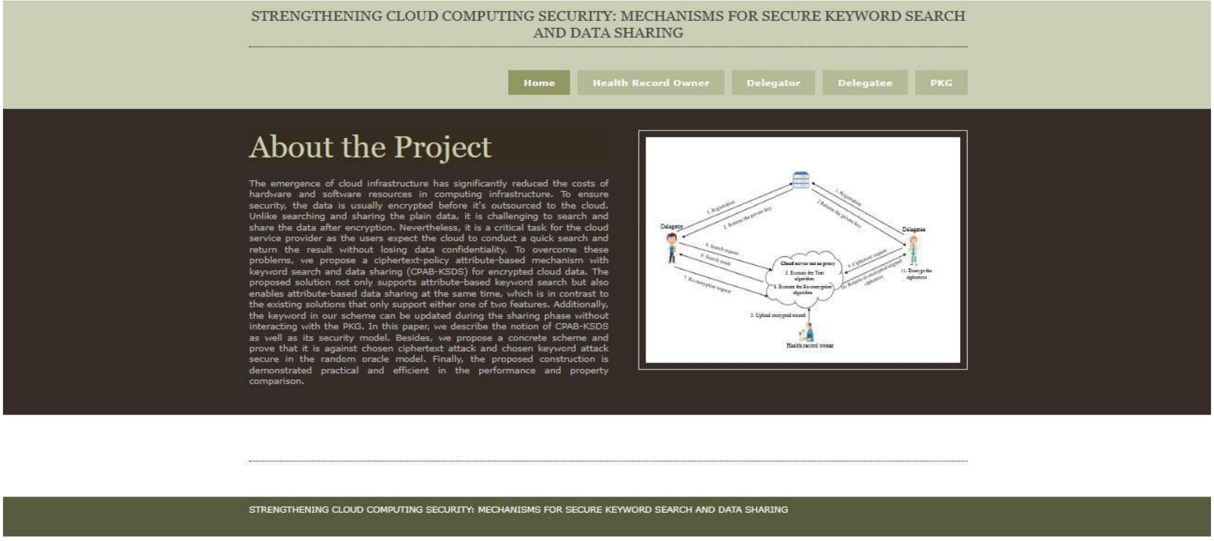
Test case 5:

Test case for PKG login:

When the PKG used to login he/she can enter username password to view all users p to view all documents, view all transactions and etc , then this results in displaying an cloud menu.

FUNCTION:	PKG LOGIN
EXPECTED RESULTS:	Should check if all the fields are filled by the client and saving the client to database
ACTUAL RESULTS:	Checking whether all the fields are field by client or not through validations and saving user.
LOW PRIORITY	NO
HIGH PRIORITY	YES

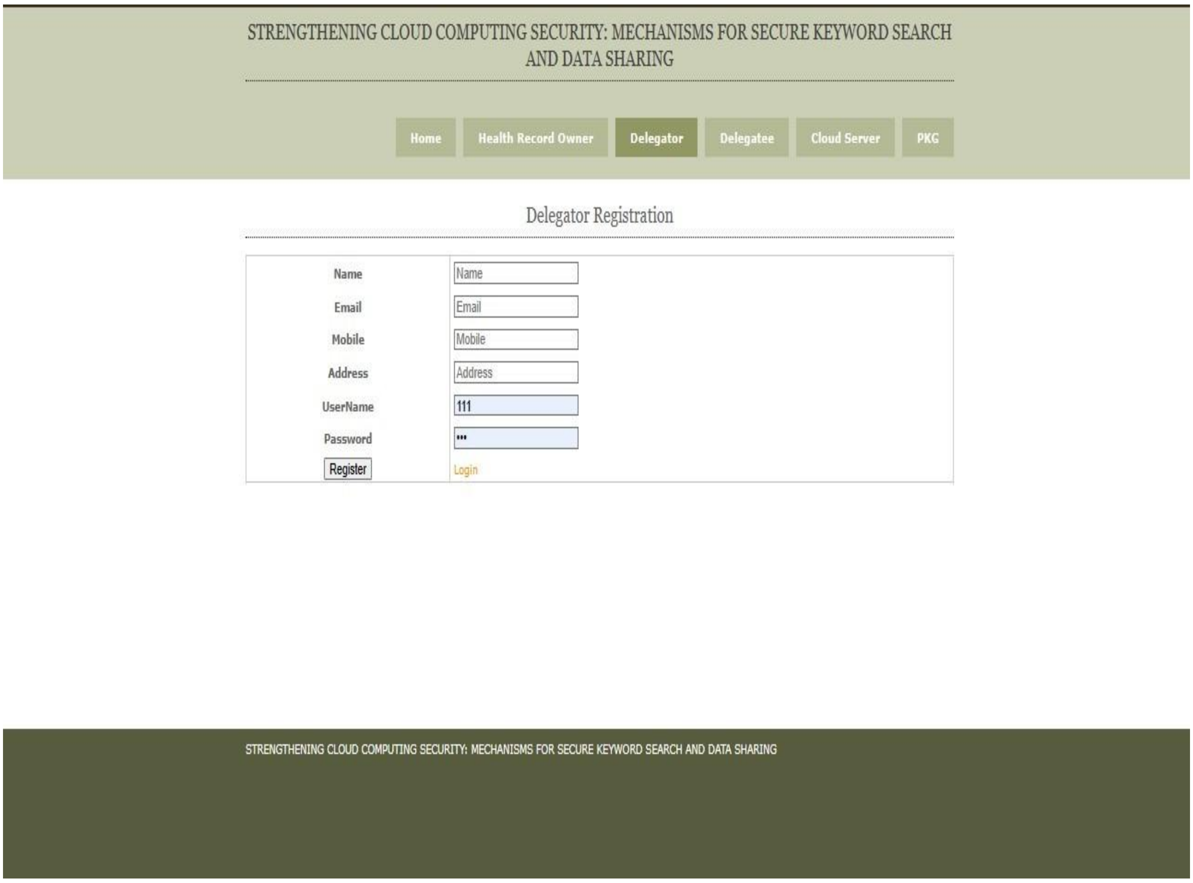
10.2 SCREENSHOTS


STRENGTHENING CLOUD COMPUTING SECURITY: MECHANISMS FOR SECURE KEYWORD SEARCH AND DATA SHARING
Home
Health Record Owner
Delegator
Delegatee
PKG
Delegator
Delegatee
1. Registration
2. Returns the private key
1. Registration
2. Returns the private key
4. Search request
6. Search result
7. Re-encryption request
Cloud server act as proxy
5. Execute the Test algorithm
8. Execute the Re-encryption algorithm
9. Ciphertext request
10. Returns re-encrypted/original ciphertext
11. Decrypt the ciphertext
3. Upload encrypted record
Health record owner
STRENGTHENING CLOUD COMPUTING SECURITY: MECHANISMS FOR SECURE KEYWORD SEARCH AND DATA SHARING

About the Project

The emergence of cloud infrastructure has significantly reduced the costs of hardware and software resources in computing infrastructure. To ensure security, the data is usually encrypted before it's outsourced to the cloud. Unlike searching and sharing the plain data, it is challenging to search and share the data after encryption. Nevertheless, it is a critical task for the cloud service provider as the users expect the cloud to conduct a quick search and return the result without losing data confidentiality. To overcome these problems, we propose a ciphertext-policy attribute-based mechanism with keyword search and data sharing (CPAB-KSDS) for encrypted cloud data. The proposed solution not only supports attribute-based keyword search but also enables attribute-based data sharing at the same time, which is in contrast to the existing solutions that only support either one of two features. Additionally, the keyword in our scheme can be updated during the sharing phase without interacting with the PKG. In this paper, we describe the notion of CPAB-KSDS as well as its security model. Besides, we propose a concrete scheme and prove that it is against chosen ciphertext attack and chosen keyword attack secure in the random oracle model. Finally, the proposed construction is demonstrated practical and efficient in the performance and property comparison.

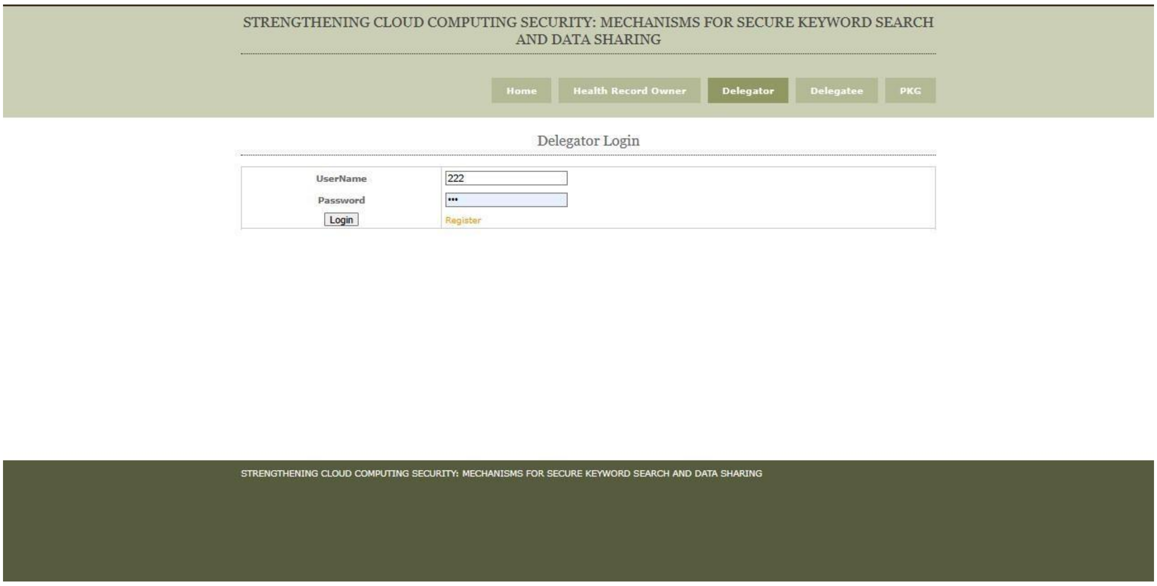
FIG-1 Home page


STRENGTHENING CLOUD COMPUTING SECURITY: MECHANISMS FOR SECURE KEYWORD SEARCH AND DATA SHARING
Home
Health Record Owner
Delegator
Delegatee
Cloud Server
PKG
Register
Login
STRENGTHENING CLOUD COMPUTING SECURITY: MECHANISMS FOR SECURE KEYWORD SEARCH AND DATA SHARING

Delegator Registration

Name	Name
Email	Email
Mobile	Mobile
Address	Address
UserName	111
Password	•••

FIG-2 Delegator registration



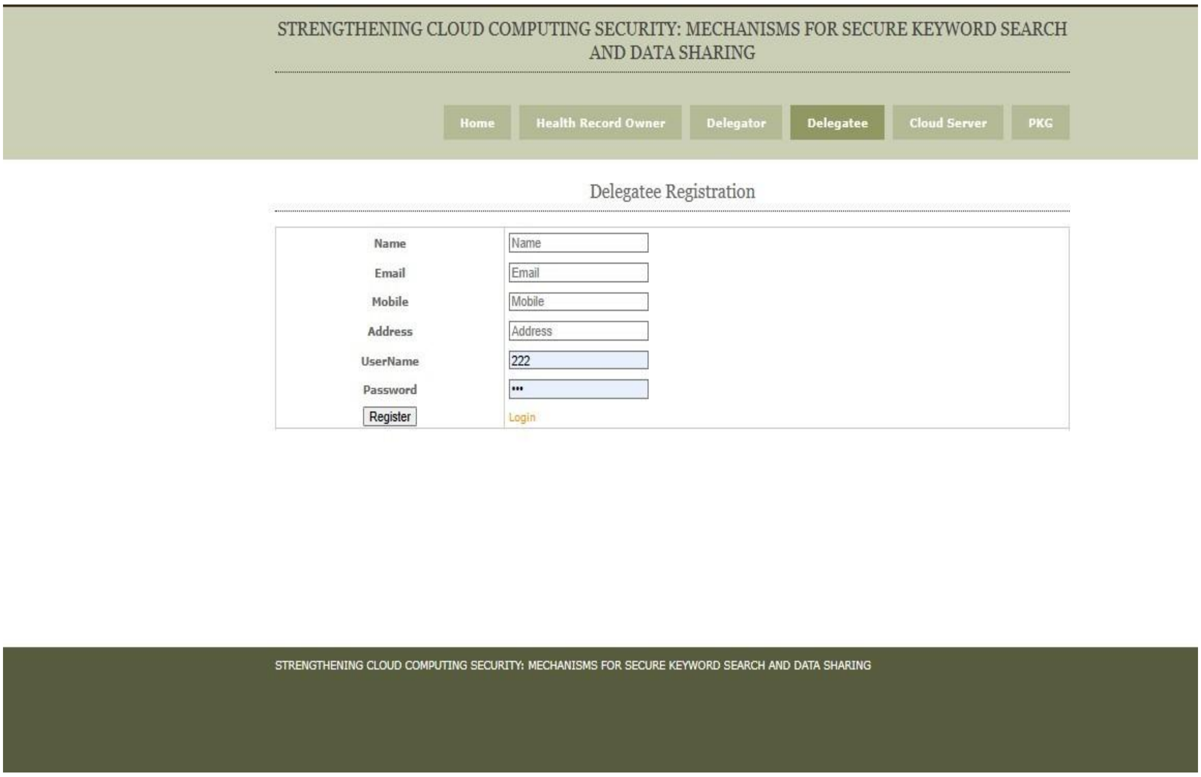
STRENGTHENING CLOUD COMPUTING SECURITY: MECHANISMS FOR SECURE KEYWORD SEARCH AND DATA SHARING

Delegator Login

UserName	222
Password	•••
Login	Register

STRENGTHENING CLOUD COMPUTING SECURITY: MECHANISMS FOR SECURE KEYWORD SEARCH AND DATA SHARING

FIG-2 Delegator login



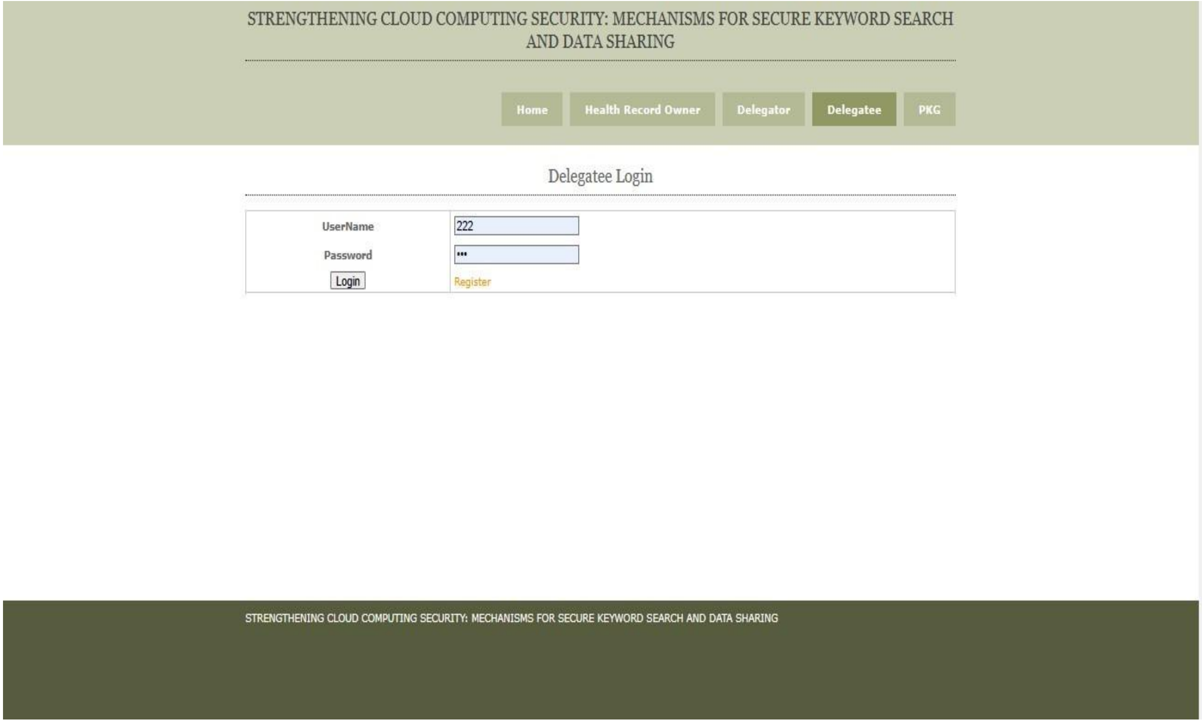
STRENGTHENING CLOUD COMPUTING SECURITY: MECHANISMS FOR SECURE KEYWORD SEARCH AND DATA SHARING

Delegatee Registration

Name	Name
Email	Email
Mobile	Mobile
Address	Address
UserName	222
Password	•••
Register	Login

STRENGTHENING CLOUD COMPUTING SECURITY: MECHANISMS FOR SECURE KEYWORD SEARCH AND DATA SHARING

FIG-3 Delegatee registration



STRENGTHENING CLOUD COMPUTING SECURITY: MECHANISMS FOR SECURE KEYWORD SEARCH AND DATA SHARING

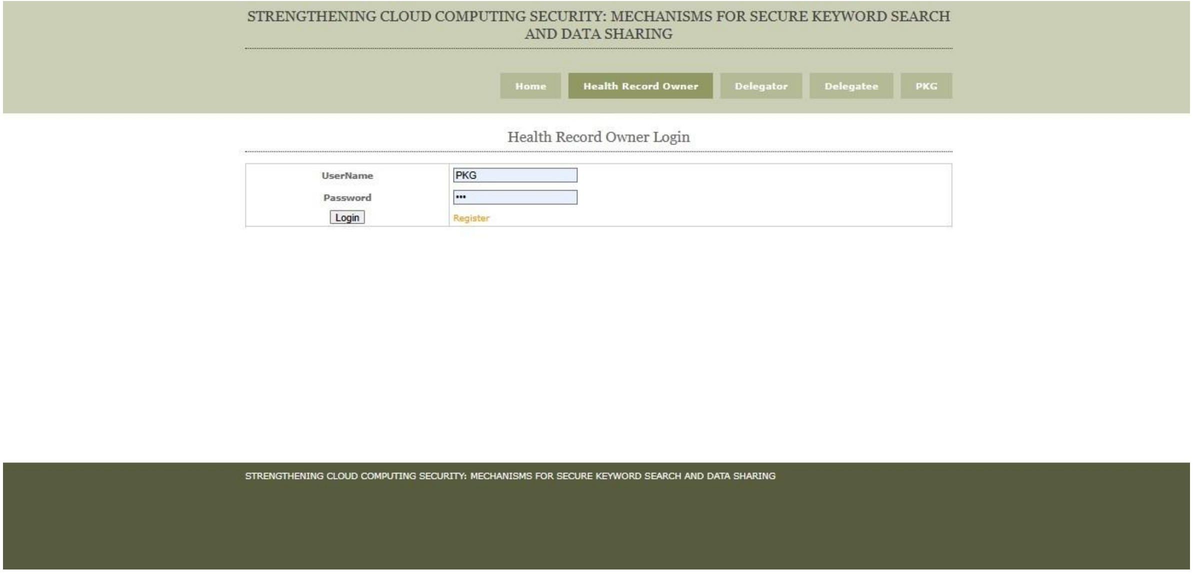
Delegatee Login

UserName: 222

Password: •••

STRENGTHENING CLOUD COMPUTING SECURITY: MECHANISMS FOR SECURE KEYWORD SEARCH AND DATA SHARING

FIG-4 Delegatee login



STRENGTHENING CLOUD COMPUTING SECURITY: MECHANISMS FOR SECURE KEYWORD SEARCH AND DATA SHARING

Health Record Owner Login

UserName: PKG

Password: •••

STRENGTHENING CLOUD COMPUTING SECURITY: MECHANISMS FOR SECURE KEYWORD SEARCH AND DATA SHARING

FIG-5 Health record owner login:

11. CONCLUSION

11.1 CONCLUSION

In this work, a new notion of ciphertext-policy attribute-based mechanism (CPAB-KSDS) is introduced to support keyword searching and data sharing. A concrete CPAB-KSDS scheme has been constructed in this paper and we prove its CCA security in the random oracle model. The proposed scheme is demonstrated efficient and practical in the performance and property comparison. This paper provides an affirmative answer to the open challenging problem pointed out in the prior work [36], which is to design an attribute-based encryption with keyword searching and data sharing without the PKG during the sharing phase. Furthermore, our work motivates interesting open problems as well including designing CPAB-KSDS scheme without random oracles or proposing a new scheme to support more expressive keyword search.

11.2 FUTURE SCOPE

The future scope of Java technology remains promising as it continues to evolve and adapt to the ever-changing landscape of software development. With the growing importance of cloud computing, microservices architecture, and the Internet of Things (IoT), Java's platform independence, strong security features, and scalability make it well-suited for modern application development. Additionally, advancements such as the modularization introduced in Java 9 and ongoing updates ensure that Java stays relevant in emerging technologies. Its widespread use in enterprise environments, coupled with a vibrant developer community and extensive ecosystem, positions Java to play a significant role in shaping the future of software development. t appears there might be a slight error in your question; "EPARA" doesn't seem to be a widely recognized term or acronym in the context of cloud computing or security. If you could provide more information or clarification about "EPARA," I would be better able to tailor my response to your specific inquiryHowever, assuming you're referring to enhancing security mechanisms in cloud computing environments, here are some general considerations and trends that might be relevant:

12. REFERENCES

1. Kai Zhang, Ximeng Liu, Yanping Li, Tao Zhang, Shuhua Yang, "A Secure Enhanced Key- Policy Attribute-Based Temporary Keyword Search Scheme in the Cloud", Access IEEE, vol. 8, pp. 127845-127855, 2020.

2.Hao Yan, Wenming Gui, "Efficient Identity-Based Public Integrity Auditing of Shared Datain Cloud Storage With User Privacy Preserving", Access IEEE, vol. 9, pp. 45822-45831, 2021.

3.Hua Shen, Mingwu Zhang, Hao Wang, Fuchun Guo, Willy Susilo, "Efficient and Privacy- Preserving Massive Data Processing for Smart Grids", Access IEEE, vol. 9, pp. 70616- 70627, 2021.

4.Jianfei Sun, Dajiang Chen, Ning Zhang, Guowen Xu, Mingjian Tang, Xuyun Nie, Mingsheng Cao, "A Privacy-Aware and Traceable Fine-Grained Data Delivery System in Cloud-Assisted Healthcare IIoT", Internet of Things Journal IEEE, vol. 8, no. 12, pp. 10034- 10046, 2021.

5.Mingwu Zhang, Yu Chen, Jiajun Huang, "SE-PPFM: A Searchable Encryption Scheme Supporting Privacy-Preserving Fuzzy Multikeyword in Cloud Systems", Systems Journal IEEE, vol. 15, no. 2, pp. 2980-2988, 2021.