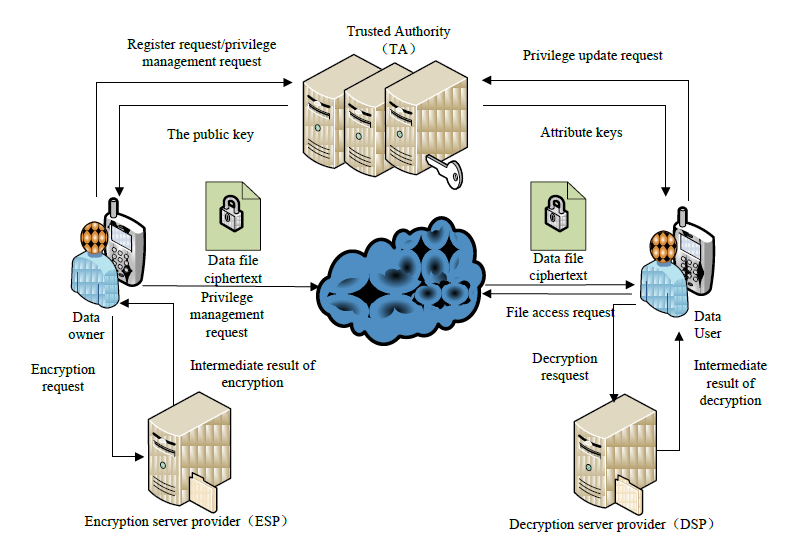
**SYSTEM DESIGN**

**SYSTEM ARCHITECTURE:**

****

**DATA FLOW DIAGRAM:**

1. The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.
2. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.
3. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.
4. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.

Unauthorized user

Owner Login

Check

No

Yes

Give Public Key Request

Upload Files

View User’s File Access Request

Send Decrypt key

Delete User’s File Access Request

End process

Revoke User’s File Access Request

View Uploaded Files

No

Unauthorized user

User Login

Check

No

Yes

Give Attribute Key Request

Attribute Key Verification

View Uploaded Files

Give File Decrypt key Request

Download File

End process

View File Download History

Unauthorized user

Trusted Authority Login

Check

No

Yes

Response Public key Requests

Response Attribute Key Request

View Uploaded Files

View Data Owner Details

View Data User Details

End process

View File and its Access Privilege

Unauthorized user

Cloud Login

Check

No

Yes

View Files and its Access Privileges

Response File Decrypt Key Requests

View File Download History

End process

**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 users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

**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.

Data User

Cloud

Trusted Authority

Data Owner

**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. It explains which class contains information.

View Data User and Data Owner Details ()

Response Public Key Requests ()

Response Attribute Key Request ()

View File and its Access Privilege ()

Trusted Authority

Login

View File and its Access Privilege ()

Response File Decrypt Key Requests ()

View File Downloaded History ()

Cloud

Login

Give Request for Attribute Key ()

Attribute Key Verification ()

View File and its Access Privilege ()

Give Request for File Decrypt Key()

Download File ()

View File Download History ()

Data User

Login

Give Request for Public Key ()

Upload Files ()

Response File Decrypt Key Requests ()

Delete and Revoke the Decrypt key Request ()

View Uploaded File and its Access Privileges ()

Data Owner

Login

**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.

SERVICE

DU

DATA BASE

TA

Cloud

Give Pubic Key Request

Give Attribute Key Request

Response Public Key and Attribute Key Requests

Upload File

View Upload File and Access privilege

Attribute Key Verification

View Files and Give Request for File Decrypt Key

Response for File Decrypt Key Request

Response for File Decrypt Key Requests

Download File

View Data Owner and Data User Details

View Files and its Access Privilege

View Files Download History

DO

Delete and Revoke the Decrypt key Request

View Files Download History

View Files and its Access Privilege

**ACTIVITY DIAGRAM:**

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

DU

TA

LOGIN

LOGIN

Start

Give Request for Attribute Key, Attribute Key Verification, View File and its Access Privilege, Give Request for File Decrypt Key, Download File, View File Download History

View Data User and Data Owner Details, Response Public Key Requests, Response Attribute Key Request, View File and its Access Privilege

Cloud

LOGIN

View File and its Access Privilege, Response File Decrypt Key Requests, View File Downloaded History

DO

LOGIN

Give Request for Public Key, Upload Files, Response File Decrypt Key Requests, Delete and Revoke the Decrypt key Request, View Uploaded File and its Access Privileges