

PROFESSIONAL TRAINING REPORT

at

**Sathyabama Institute of Science and Technology
(Deemed to be University)**

Submitted in partial fulfillment of the requirements for the
award of Bachelor of Engineering in Computer Science
And Engineering

By

**Gopalachetty vedanth
(REG. NO. 38110168)**

**Channu harsha vardhan reddy
(REG.NO. 38110198)**



**DEPARTMENT OF COMPUTER SCIENCE AND
ENGINEERING**

SCHOOL OF COMPUTING

**SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY
JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI – 600119, TAMILNADU**



**SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited with Grade "A" by NAAC
(Established under Section 3 of UGC Act,
1956) JEPPIAAR NAGAR, RAJIV GANDHI
SALAI

CHENNAI- 600119

www.sathyabama.ac.in



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **Vedanth.G** (Reg. No: 38110168) Harsha vardhan reddy (Reg. No: 38110198) who carried out the project entitled **"DATA SECURITY AND PRIVACY PROTECTION FOR CLOUD STORAGE USING BCH"** under my supervision from June 2021 to November 2021.

Internal Guide

MS. R. Yogitha, M.E.,

Ph.D.,

Head of the Department

Dr. S. VIGNESHWARI, M.E., PH.D.,

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

I **G.VEDANTH (REG.NO:38110168), CHANNU HARSHA VARDHAN REDDY (REG.NO: 38110198)** hereby declare that the project report entitled **Data security And Privacy Protection for Cloud Storage Using BCH** done by me under the guidance of **MS. R. YOGITHA** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering Degree in Computer Science and Engineering.

DATE: 01-03-2022

PLACE: CHENNAI

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala M.E., Ph.D, Dean, School of Computing, Dr. S. Vigneshwari M.E., Ph.D., and Dr. L. Lakshmanan M.E., Ph.D., Head of the Department of Information Technology** for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **MS. R. YOGITHA M.E., Ph.D.**, for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science And Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

In recent technology cloud storage technology which is gaining more and more attention.. The protection of the secrecy of the arts relies for the most part on the fact that encryption technology while storing or processing the customer data. Our cloud-based, three-layer storage framework. The proposed framework can use cloud storage and protect secret information. The hash-Solomon coding algorithm is created for distinct portions, and it using data to divide it. The first part received the knowledge, if the knowledge is lacking, and it have lost. In the real world from the human point of view of which are the algorithms secure the data and use the knowledge and assurance of a bucket, and in accordance with what has been designed by the effective cause cannot be in the reason Furthermore, SaaS, According to the computational intelligence algorithm that can compute distribution clouds, fog, and machine, respectively, SaaS , the customer rejected their request in the accessible hosting environment via the network from various clients via application users. The customer was unable or unwilling to submit them to the controlling cloud infrastructure, especially in cloud settings like, with the exception of limited user-specific application configuration. For SaaS models, Google Apps and Microsoft Office 365 are good options.

Table of Contents

ACKNOWLEDGEMENT	i
DECLARATION	ii
ABSTRACT.....	iii
Chapter -1.....	1
1. Introduction.....	1
1.2 Existing System.....	3
1.3 Proposed System	3
Chapter -2.....	4
2. Literatute Survey	4
Chapter -3.....	7
Analysis.....	7
3.1 Introduction	7
3.2 Feasibility Study	7
3.2.1 Hardware Requirements	9
3.2.2 Software Requirements.....	9
Chapter -4.....	10
4. Implementataion	11
4.1 System Design.....	19
Chapter-5.....	22
5.1 Introduction	23
5.1 MODULE DESCRIPTION.....	23
5.1.1 LOGIN MODULE.....	23
5.1.2 REGISTRATION MODULE.....	23
5.1.3 STORAGE MODULE.....	24
5.1.4 RECOVERY MODULE.....	24
Chapter -6.....	24

6.	Testing	21
	6.1 UnitTesting.....	23
	6.2 Integration Testing	23
	6.3 Functional Test.....	23
	6.4 System Test	23
	Chapter -7.....	28
7.	Results and Discussion	28
	Chapter -8.....	29
8.	Conclusion	29
	Chapter -9.....	30
9.	Refference	31
	Chapter-10.....	33
10.	Appendix.....	34
	Chapter -11.....	41
11.1	Excussion Process.....	41
11.2	SOURCE CODE.....	42

List Of Screens

S. No.	Figure No.	Screen Name	Page No.
01	11.1	Home Page	34
02	11.2	Login page	35
03	11.3	User Image	35
04	11.4	File image	36
05	11.5	encryption	37
06	11.6	Decryption	38
07	11.7	Combine data	39

CHAPTER - 1

INTRODUCTION

Cloud computing is a term used in the IT industry to denote a form of IT outsourcing service, similar to how energy is outsourced. Users are limited to using just. You don't need to say where the anxiety caused by electricity, how it occurred, or to transfer. For each day of the month, give it back. After cloud computing, the idea is similar to: The user can only use storage, computing power, is, or has been called upon with the development of the profession, for them is a work of the anguish of a freedom of those on the outside. Cloud is the Internet computer network diagram metaphor for how the Internet is depicted; how the removal of everything is hiding in the Internet infrastructure. There is a way in which the related count developers are "allowing users of Internet access technology (" cloud ") service is not knowledge or power technologies after servers.

And that the clouds with a great mist can be perceived and the means to count it and to give itself to the works, which is increased by the flow for an example of the difficulty of the thing [1]. The effects of fog on a large computer and computer system vary. However, a frequent approach that can be extracted is a limitation in the distribution of accurate material, and the issue is that developing and reviewing metrics to increase accuracy is difficult. Fog is networked via plane and data control plan in the data plane, for example, cloud computing allows metering to take place at the edge of network information services rather than on central servers. Compared to cloud computing, cloud computing emphasizes proximity to end users and a customer for some, dense geographic distribution and the pooling of local resources, latency for people and savings in bandwidth for mining edge analytics / flow; so that it becomes the top redundancy and user experience, it can only be, even if they did not use the AAL to pass, that the lack of missions.

To protect user privacy, we offer a framework according to the TLS cloud computing model. TSL is a power user management framework that can effectively protect user privacy. With which the interiors were barely able to attack. Traditional approaches to work without, and so on, in the attacks, but with the CSP is proportional to the difficulties, all the traditional ways of being null. In different traditional ways in our program, user data is divided into three sections with encoding technology of different sizes. Between North Korea and some of the

key information to the abundance of all things except on their way. The data from the computer model fog will be stored in the cloud server in order of small to large. And recovered by an opponent of old age, if the knowledge is not certain knowledge in this way, it adorns the whole User. CSP and that it will not be useful information Without a cloud server and the name of the local erver and whatever machine at the time of the world, and most certainly, only the cloud of users. The framework can make full use of cloud storage while maintaining data privacy. Cloud computing has piqued the interest of many people from various walks of life. The three-layer cloud storage divides data into three parts, and if one of these parts is missing, the data is lost. The bucket concept-based algorithms are used in this proposed framework. In this paper, we discuss the challenges of deep learning security and privacy.

EXISTING SYSTEM

- The computer technology has developed rapidly. Cloud computing has gradually matured through so many people effort's.
- In current storage schema, the user's data is totally stored in cloud servers. If the user lose their right of control on data and face privacy risk.
- The privacy protection schemes are usually based on encryption technology. These kinds of methods cannot effectively resist attack from the inside of cloud server.

PROPOSED SYSTEM

- The framework can take full of cloud storage and protect the privacy of data.
- Here the cloud computing has attracted great attention from different sector of society.
- The three layer cloud storage stores in to the three different of data parts .
- If the one data part missing we lost the data information. In this proposed framework using the bucket concept based algorithms

CHAPTER-2

LITERATURE SURVEY

A new secret preserving cloud services. Our solution is based on the nature of the signature for some of the anonymous in accessing cloud storage services to a group of agents that are not bilinear and that they are shared servers. Offers a new solution for anonymous authentication of registered users. It can be demonstrated without revealing the identity of users and cloud users can use their services without any profiling of threat behavior. However, if the user's provider violates the access right to be revoked. Our solution gives access to anonymous, and is transmitted to the knowledge of the secret of the possibility of dissociating. The proof-of-concept application to implement our solution and the results of this experiment. In addition, privacy solutions for cloud services execute a foolproof rule, and group signing of core skills to the role of privacy enhancing solutions for cloud services. From there, compare our performance with associated solutions and diagrams.. Secret plans are the most widely used type of metric definition code, that is, an error correction code defining a forbidden space by prescription, increasing or decreasing the monotonic set. Then redefine to a new problem package, and the settings fix these errors is usually an opportunity to fix the error, but some code (taking into account this fresh metric and new package).

It brings to light the problems of the disciples, and of the intellectual obfuscation, the finance and insurance industries are confidential information belonging to the name of the. The secret of authoritarian times is information that is hidden, so the danger of misuse of the thing. To be cut out and given to us in the name of the third part of the Software by the services of digital steel its heart and close to its mismanagement causing a preventive breach of confidentiality from the cloud, where enormous amounts of data are stored and maintained. Protecting user privacy is a major concern in today's cloud computing world. even with changes in the field of cloud computing to improve its efficiency, efficiency and optimization among developers. Office, etc., information reliability and cloud user identification and maintainability CPs.. It is done as stated in the user's own information and today's technology is more related to my heart, do they

work. No, but they have the cloud provider issue that is most important when it comes to data suggestions for cloud and digital data storage that you do, this way and that, and keep it. , worldwide. The reason for the need for the issues proposed in cloud computing research. The digital data recorded in the cloud led us to Prevent Loss Model's Privacy offering (ppm, dDLC). This proposal aids the CR's confidence in proprietary data and cloud-based data. It is simple and its Internet environment can provide users, such as various offices, Software-in-Service, Order-at-Service, the risk of changes in the accuracy of the order of. It represents a consumer technology divided between various arts like Porta integrity of the audit mechanism.. For-use-to-service the security of data with the users interacts and organizations cannot store their data in the cloud, the new commander, plaster encoded form, Merkle-hash-tree, construction. This secure data storage and an efficient storage cloud. This article discusses privacy and security architectures in day-to-day management in cloud storage.. Later trying to kill the participation of new information on verifiable secret plans (vss). Define a new one for us, "metric" (the properties look slightly different from a standard Hamming meter). Error correction technique, of which it is well known that the VSS and distributed commitments in pairs by holding the test protocol and the error correction capability of the error correction codes establish interleaving.

Internet of Things (IoT), smart cities, business digital transformation, and the global digital economy are among the newest emerging trends. On account of the huge data created, the continual expansion of data storage pressure drives the rapid development of the whole storage industry. Responsibility for business continuity in the digital secrecy project and in widely dispersed regions of the world, and close to its mismanagement causing a preventive breach of confidentiality from the cloud . Security vulnerabilities from the past still exist in cloud computing systems. Traditional security techniques are no longer adequate for cloud apps and data as business boundaries have been stretched to the cloud. Cloud computing is having a huge influence on the subject of information security due to its openness and multi-tenant nature.. Later look at a number of data storage security and privacy challenges and techniques in the cloud computing environment in this study the techniques employed in Cloud computing, where data

security is a vital component. Integrity, confidentiality, and accessibility of data are all important considerations. As a result, personal data privacy issues and cloud-related technologies are being investigated. Data security and privacy have long been intertwined. By safeguarding data in the cloud, comparative data security and privacy studies may assist enhance user confidence..

CHAPTER - 3 ANALYSIS

3.1 INTRODUCTION:

We design a three-layer storage framework based on fog computing. The proposed framework can both take full advantage of cloud storage and protect the privacy of data. Here we are using Hash-Solomon code algorithm is designed to divide data into different parts. If the one data part missing we lost the data information. In this framework we are using bucket concept based algorithms and secure the data information and then it can show the security and efficiency in our scheme. Moreover, based on computational intelligence, this algorithm can compute the distribution proportion stored in cloud 80% of data, and fog 15% of data, and local machine 5% of data, respectively. The framework can take full of cloud storage and protect the privacy of data. Here the cloud computing has attracted great attention from different sector of society. The three layer cloud storage stores in to the three different parts of data parts .If the one data part missing we lost the data information. In this proposed framework using the bucket concept based algorithms. In our system we using a bucket concept so reduce the data wastages and reduce the process timings. We are using a BCH (Bose–Chaudhuri–Hocquenghem) code algorithm. It's High flexible. BCH code are used in many communications application and low amount of redundancy.

3.2 FEASIBILITY STUDY:

The feasibility of the project is analyzed in this phase and business proposal is put forth with a 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 understanding of the major requirements for the system is essential.

The feasibility study investigates the problem and the information needs of the stakeholders. It seeks to determine the resources required to provide an information

systems solution, the cost and benefits of such a solution, and the feasibility of such a solution.

The goal of the feasibility study is to consider alternative information systems solutions, evaluate their feasibility, and propose the alternative most suitable to the organization. The feasibility of a proposed solution is evaluated in terms of its components.

1 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. The expenditures must be justified. Thus the developed system is 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.

2 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 no changes are required for implementing this system.

3 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.

a. SOFTWARE REQUIREMENTS& HARDWARE REQUIREMENTS:

HARDWARE REQUIREMENTS:

- System - Pentium-IV
- Speed - 2.4GHZ
- Hard disk - 40GB
- Monitor - 15VGA color
- RAM - 512MB

SOFTWARE REQUIREMENTS:

- Operating System - Windows XP
- Coding language - .Net
- IDE - Visual Studio 2013
- Database -SQL Server

CHAPTER-4

4.IMPLEMENTATION

INTRODUCTION TO .NET

FEATURES OF .NET

Microsoft .NET is a set of Microsoft software technologies for rapidly building and integrating XML Web services, Microsoft Windows-based applications, and Web solutions. The .NET Framework is a language-neutral platform for writing programs that can easily and securely interoperate. There's no language barrier with .NET: there are numerous languages available to the developer including Managed C++, C#, Visual Basic and Java Script. The .NET framework provides the foundation for components to interact seamlessly, whether locally or remotely on different platforms. It standardizes common data types and communications protocols so that components created in different languages can easily interoperate.

“.NET” is also the collective name given to various software components built upon the .NET platform. These will be both products (Visual Studio.NET and Windows.NET Server, for instance) and services (like Passport, .NET My Services, and so on).

2.2.1 THE .NET FRAMEWORK

The .NET Framework has two main parts:

1. The Common Language Runtime (CLR).
2. A hierarchical set of class libraries.

The CLR is described as the “execution engine” of .NET. It provides the environment within which programs run. The most important features are

- ◆ Conversion from a low-level assembler-style language, called Intermediate Language (IL), into code native to the platform being executed on.
- ◆ Memory management, notably including garbage collection.
- ◆ Checking and enforcing security restrictions on the running code.

- ◆ Loading and executing programs, with version control and other such features.
- ◆ The following features of the .NET framework are also worth description:

Managed Code

The code that targets .NET, and which contains certain extra information - "metadata" - to describe itself. Whilst both managed and unmanaged code can run in the runtime, only managed code contains the information that allows the CLR to guarantee, for instance, safe execution and interoperability.

Managed Data

With Managed Code comes Managed Data. CLR provides memory allocation and deallocation facilities, and garbage collection. Some .NET languages use Managed Data by default, such as C#, Visual Basic.NET and JScript.NET, whereas others, namely C++, do not. Targeting CLR can, depending on the language you're using, impose certain constraints on the features available. As with managed and unmanaged code, one can have both managed and unmanaged data in .NET applications - data that doesn't get garbage collected but instead is looked after by unmanaged code.

Common Type System

The CLR uses something called the Common Type System (CTS) to strictly enforce type-safety. This ensures that all classes are compatible with each other, by describing types in a common way. CTS define how types work within the runtime, which enables types in one language to interoperate with types in another language, including cross-language exception handling. As well as ensuring that types are only used in appropriate ways, the runtime also ensures that code doesn't attempt to access memory that hasn't been allocated to it.

Common Language Specification

The CLR provides built-in support for language interoperability. To ensure that you can develop managed code that can be fully used by developers using any programming language, a set of language features and rules for using them called the

Common Language Specification (CLS) has been defined. Components that follow these rules and expose only CLS features are considered CLS-compliant.

2.2.2 THE CLASS LIBRARY

.NET provides a single-rooted hierarchy of classes, containing over 7000 types. The root of the namespace is called System; this contains basic types like Byte, Double, Boolean, and String, as well as Object. All objects derive from System. Object. As well as objects, there are value types. Value types can be allocated on the stack, which can provide useful flexibility. There are also efficient means of converting value types to object types if and when necessary.

The set of classes is pretty comprehensive, providing collections, file, screen, and network I/O, threading, and so on, as well as XML and database connectivity.

The class library is subdivided into a number of sets (or namespaces), each providing distinct areas of functionality, with dependencies between the namespaces kept to a minimum.

2.2.3 LANGUAGES SUPPORTED BY .NET

The multi-language capability of the .NET Framework and Visual Studio .NET enables developers to use their existing programming skills to build all types of applications and XML Web services. The .NET framework supports new versions of Microsoft's old favorites Visual Basic and C++ (as VB.NET and Managed C++), but there are also a number of new additions to the family.

Visual Basic .NET has been updated to include many new and improved language features that make it a powerful object-oriented programming language. These features include inheritance, interfaces, and overloading, among others. Visual Basic also now supports structured exception handling, custom attributes and also supports multi-threading.

Visual Basic .NET is also CLS compliant, which means that any CLS-compliant language can use the classes, objects, and components you create in Visual Basic .NET.

Managed Extensions for C++ and attributed programming are just some of the enhancements made to the C++ language. Managed Extensions simplify the task of migrating existing C++ applications to the new .NET Framework. C# is Microsoft's new

language. It's a C-style language that is essentially "C++ for Rapid Application Development". Unlike other languages, its specification is just the grammar of the language. It has no standard library of its own, and instead has been designed with the intention of using the .NET libraries as its own.

Microsoft Visual J# .NET provides the easiest transition for Java-language developers into the world of XML Web Services and dramatically improves the interoperability of Java-language programs with existing software written in a variety of other programming languages.

Active State has created Visual Perl and Visual Python, which enable .NET-aware applications to be built in either Perl or Python. Both products can be integrated into the Visual Studio .NET environment. Visual Perl includes support for Active State's Perl Dev Kit.

Other languages for which .NET compilers are available include

- FORTRAN
- COBOL
- Eiffel

ASP.NET XML WEB SERVICES	Windows Forms
Base Class Libraries	
Common Language Runtime	
Operating System	

Figure. NET Framework

C#.NET is also compliant with CLS (Common Language Specification) and supports structured exception handling. CLS is set of rules and constructs that are supported by the CLR (Common Language Runtime). CLR is the runtime environment provided by the .NET Framework; it manages the execution of the code and also makes the development process easier by providing services C#.NET is a CLS-compliant language. Any objects, classes, or components that created in C#.NET can be used in

any other CLS-compliant language. In addition, we can use objects, classes, and components created in other CLS-compliant languages in C#.NET .The use of CLS ensures complete interoperability among applications, regardless of the languages used to create the application.

Constructors and Destructors

Constructors are used to initialize objects, whereas destructors are used to destroy them. In other words, destructors are used to release the resources allocated to the object. In C#.NET the sub finalize procedure is available. The sub finalize procedure is used to complete the tasks that must be performed when an object is destroyed. The sub finalize procedure is called automatically when an object is destroyed. In addition, the sub finalize procedure can be called only from the class it belongs to or from derived classes.

Garbage Collection

Garbage Collection is another new feature in C#.NET. The .NET Framework monitors allocated resources, such as objects and variables. In addition, the .NET Framework automatically releases memory for reuse by destroying objects that are no longer in use. In C#.NET, the garbage collector checks for the objects that are not currently in use by applications. When the garbage collector comes across an object that is marked for garbage collection, it releases the memory occupied by the object.

Overloading

Overloading is another feature in C#. Overloading enables us to define multiple procedures with the same name, where each procedure has a different set of arguments. Besides using overloading for procedures, we can use it for constructors and properties in a class.

Multithreading

C#.NET also supports multithreading. An application that supports multithreading

can handle multiple tasks simultaneously, we can use multithreading to decrease the time taken by an application to respond to user interaction.

Structured Exception Handling

C#.NET supports structured handling, which enables us to detect and remove errors at runtime. In C#.NET, we need to use Try...Catch...Finally statements to create exception handlers. Using Try...Catch...Finally statements, we can create robust and effective exception handlers to improve the performance of our application.

The .Net Framework

The .NET Framework is a new computing platform that simplifies application development in the highly distributed environment of the Internet.

2.2.4 OBJECTIVES OF .NET FRAMEWORK

1. To provide a consistent object-oriented programming environment whether object codes is stored and executed locally on Internet-distributed, or executed remotely.
2. To provide a code-execution environment to minimizes software deployment and guarantees safe execution of code.
3. Eliminates the performance problems.

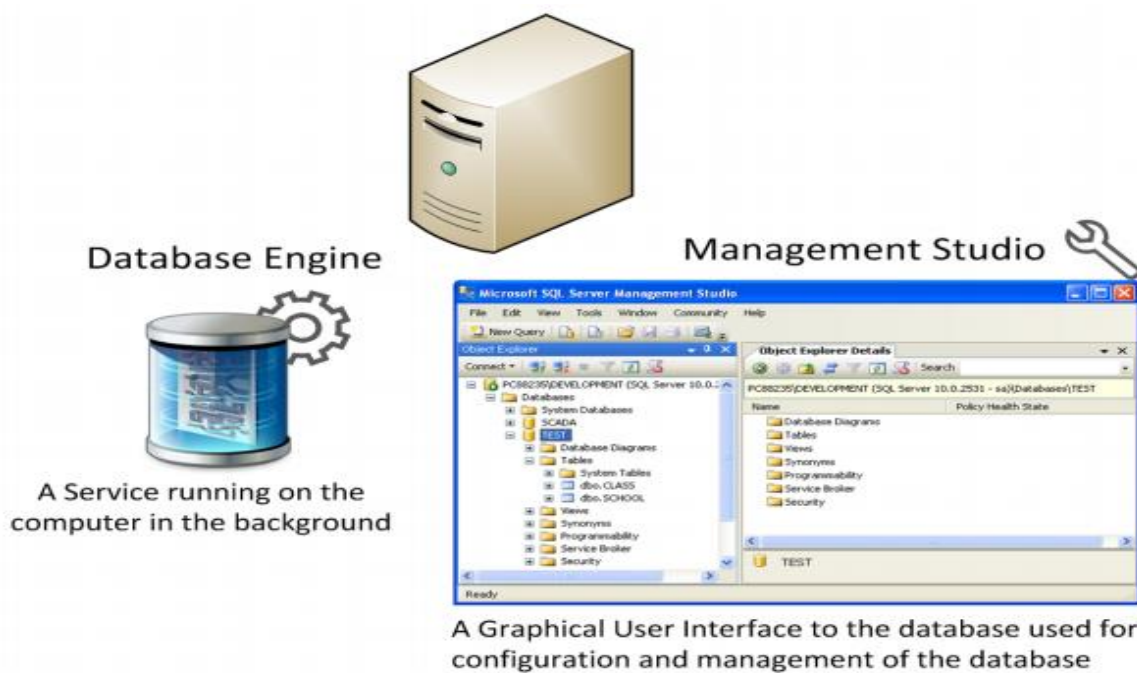
There are different types of application, such as Windows-based applications and Web-based applications.

2.3 SQL SERVER

Microsoft is the vendor of SQL Server. We have different editions of SQL Server, where SQL Server Express is free to download and use. SQL Server uses T-SQL (Transact-SQL). T-SQL is Microsoft's proprietary extension to SQL. TSQL is very similar to standard SQL, but in addition it supports some extra functionality, built in functions, etc.

T-SQL expands on the SQL standard to include procedural programming, local variables, and various support functions for string processing, date processing, mathematics, etc. SQL Server consists of a Database Engine and a Management Studio (and lots of other stuff which we will not mention here). The Database engine has no graphical interface - it is just a service running in the background of your computer (preferable on the server). The Management Studio is graphical tool for configuring and viewing the information in the database. It can be installed on the server or on the client (or both).

SQL Server Management Studio can also be used to create a new database, alter any existing database schema by adding or modifying tables and indexes, or analyze performance. It includes the query windows which provide a GUI based interface to write and execute queries.



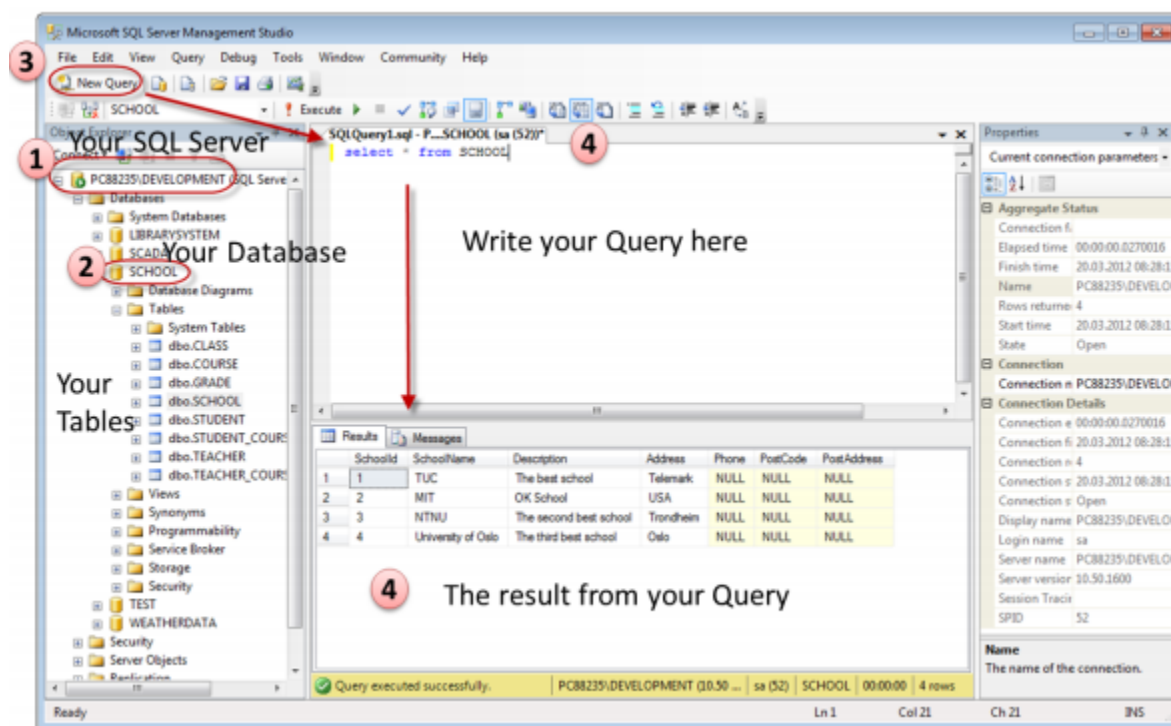
2.3.1 SQL SERVER MANAGEMENT STUDIO

SQL Server Management Studio is a GUI tool included with SQL Server for configuring, managing, and administering all components within Microsoft SQL Server. The tool includes both script editors and graphical tools that work with objects and features of the server. As mentioned earlier, version of SQL Server Management Studio is also available for SQL Server Express Edition, for which it is known as SQL Server Management Studio Express.

A central feature of SQL Server Management Studio is the Object Explorer, which allows the user to browse, select, and act upon any of the objects within the server. It can be used to visually observe and analyze query plans and optimize the database performance, among others. SQL Server Management Studio can also be used to create a new database, alter any existing database schema by adding or modifying tables and indexes, or analyze performance. It includes the query windows which provide a GUI based interface to write and execute queries.

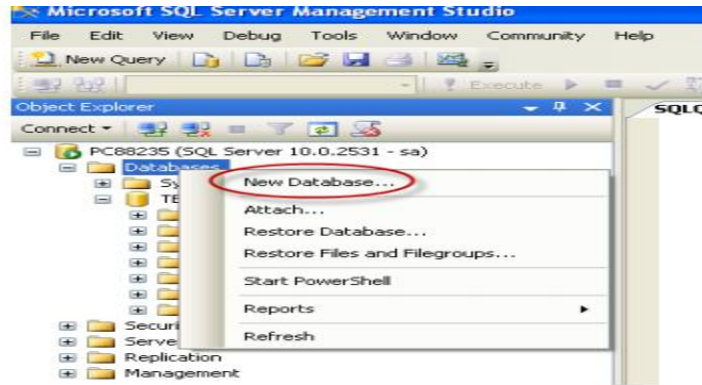
When creating SQL commands and queries, the "Query Editor" (select "New

Query” from the Toolbar) is used (shown in the figure above). With SQL and the “Query Editor” we can do almost everything with code, but sometimes it is also a good idea to use the different Designer tools in SQL to help us do the work without coding (so much).



2.3.2 Create a new Database

It is quite simple to create a new database in Microsoft SQL Server. Just right-click on the “Databases” node and select “New Database...”



The OLAP Services feature available in SQL Server version 7.0 is now called SQL Server 2000 Analysis Services. The term OLAP Services has been replaced with the term Analysis Services. Analysis Services also includes a new data mining component. The Repository component available in SQL Server version 7.0 is now called Microsoft SQL Server 2000 Meta Data Services. References to the component now use the term Meta Data Services. The term repository is used only in reference to the repository engine within Meta Data Services

Advantages

- In our system we using a bucket concept so reduce the data wastages and reduce the process timings.
- We are using a BCH (Bose–Chaudhuri–Hocquenghem) code algorithm. It's High flexible.
- BCH code are used in many communications application and low amount of redundancy.

Disadvantages

- Changes in the understanding of risk as a result of extending the data centre into the cloud.
- Low latency and location awareness

4.1 SYSTEM DESIGN

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 considered the following things:

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

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.

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

- Convey information about past activities, current status or projections of the
- Future.
- Signal important events, opportunities, problems, or warnings.
- Trigger an action.
- Confirm an action.

CHAPTER-5

5.1 INTRODUCTION:

MODULE DESCRIPTION

MODULES

- Login
- Registration
- Storage Scheme
- Recovery Scheme

Login Module

This is the first activity that opens when user open the website. User needs to provide a correct contact number and a password, which user enters while registering, in order to login into the website. If information provided by the user matches with the data in the database table then user successfully login into the website else message of login failed is displayed and user need to reenter correct information. A link to the register activity is also provided for registration of new users.

Registration Module

A new user who wants to access the webpage needs to register first before login. By clicking on register button in login activity, the register activity gets open. A new user registers by entering full name, password and contact number. A user needs to enter password again in confirm password textbox for confirmation. When user enters the information in all textboxes, on the click of register button, the data is transferred to database and user is directed to login activity again. Registered user then needs to login in order to access the webpage. Validations are applied on all the textboxes for proper functioning of the webpage. Like information in each textbox is must that is each

textbox, either it is of name, contact, password or confirm password, will not be empty while registering. If any such textbox is empty app will give message of information is must in each textbox. Also data in password and confirm password fields must match for successful registration. Another validation is contact number must be valid one that is of 10 digits. If any such validation is violated then registration will be unsuccessful and then user needs to register again. Message that webpage will display when one of the field is empty. If all such information is correct user will be directed to login activity for login into the webpage.

Storage Module

In this module, user can store their files into three different storage server. The data owner has not control over the data after it is uploaded on cloud. In this module, the original data get encrypted into three different layer's. The data in each layer can be encrypted by using different cryptographic algorithm's and encryption key before storing them in the Cloud.

Recovery Module

In this Module, user can recover their files from three different storage server that is Cloud server, Fog server and Local machine. Here we are using Hash-Solomon code algorithm is designed to divide data into different parts. If the one data part missing we lost the data information.

CHAPTER - 6

TESTING

Test plan

Software testing is the process of evaluation a software item to detect differences between given input and expected output. Also to assess the feature of a software item. Testing assesses the quality of the product. Software testing is a process that should be done during the development process. In other words software testing is a verification and validation process.

Verification

Verification is the process to make sure the product satisfies the conditions imposed at the start of the development phase. In other words, to make sure the product behaves the way we want it to.

Validation

Validation is the process to make sure the product satisfies the specified requirements at the end of the development phase. In other words, to make sure the product is built as per customer requirements.

Basics of software testing

There are two basics of software testing: black box testing and white box testing.

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 works.

ALPHATESTING:

Combination of three testing methods (unit, integration and system testing) forms the alpha testing. Alpha testing is conducted by the development team and quality assurance team. The following things in the application are tested during the alphatesting:

SpellingMistakes

BrokenLinks

CloudyDirections

Measuring the loading time in the minimum system specifications for optimization.

BETATESTING:

Once the alpha testing is completed successfully, the product development team moves onto beta testing. In beta testing, a prerelease version of the software isprovided to sample user base over the internet. These early users will use the software and provide a timely feedback about the software working. Following steps are performed in thistesting:

Installation of the software by the users and feedback submission to the development

team at regular intervals.

Noting down the visual errors, generating crash reports.

Once the feedback is received, development team fixes the issues and once all the known bugs are fixed, a final version is released to the users.

The nature of the software determines the consumer satisfaction. So, as more and more issues are fixed, nature of the product is increased which results in higher consumer satisfaction..

White box Testing

White box testing is a testing technique that takes into account the internal mechanism of a system. It is also called structural testing and glass box testing. Black box testing is often used for validation and white box testing is often used for verification.

Types of testing

There are many types of testing like

- Unit Testing
- Integration Testing
- Functional Testing
- System Testing
- Stress Testing
- Performance Testing
- Usability Testing
- Acceptance Testing
- Regression Testing
- Beta Testing

Unit Testing

Unit testing is the testing of an individual unit or group of related units. It falls under the class of white box testing. It is often done by the programmer to test that the unit he/she has implemented is producing expected output against given input.

Integration Testing

Integration testing is testing in which a group of components are combined to produce output. Also, the interaction between software and hardware is tested in integration testing if software and hardware components have any relation. It may fall under both white box testing and black box testing.

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 centered 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 is the testing to ensure that by putting the software in different environments (e.g., Operating Systems) it still works. System testing is done with full system implementation and environment. It falls under the class of black box testing.

Stress Testing

Stress testing is the testing to evaluate how system behaves under unfavorable conditions. Testing is conducted at beyond limits of the specifications. It falls under the class of black box testing.

Performance Testing

Performance testing is the testing to assess the speed and effectiveness of the system and to make sure it is generating results within a specified time as in performance requirements. It falls under the class of black box testing.

Usability Testing

Usability testing is performed to the perspective of the client, to evaluate how the GUI is user-friendly? How easily can the client learn? After learning how to use, how proficiently can the client perform? How pleasing is it to use its design? This falls under the class of black box testing.

Acceptance Testing

Acceptance testing is often done by the customer to ensure that the delivered product meets the requirements and works as the customer expected. It falls under the class of black box testing.

Regression Testing

Regression testing is the testing after modification of a system, component, or a group of related units to ensure that the modification is working correctly and is not damaging or imposing other modules to produce unexpected results. It falls under the class.

Chapter - 7

RESULTS AND DISCUSSION

Separation of sensitive data and access control are two major considerations in data security and privacy protection. Our goal is to develop a set of universal identity management and privacy protection .Not only for the cloud, but also for data bases and hard discs, will implement the same concept with diverse perspectives in the field of data security. Proposed methodology built this model security as a bch and bucket idea method in.net programming, which will be able to improve in the future utilising the easiest style of coding, python programming. Our system utilised html to track the process using buttons, which can now do with Word Press by dragging and dropping the button and triggering the process.

Chapter-8

8.CONCLUSION

The development of cloud computing brings us a lot of benefits. Cloud storage is a convenient technology which helps users to expand their storage capacity. However, cloud storage also causes a series of secure problems. When using cloud storage, users do not actually control the physical storage of their data and it results in the separation of ownership and management of data. In order to solve the problem of privacy protection in cloud storage, we propose a TLS framework based on fog computing model and design a BCH Code algorithm. Through the theoretical safety analysis, the scheme is proved to be feasible. By allocating the ratio of data blocks stored in different servers reasonably, we can ensure the privacy of data in each server. On another hand, cracking the encoding matrix is impossible theoretically. Besides, using hash transformation can protect the fragmentary information. Through the experiment test, this scheme can efficiently complete encoding and decoding without influence of the cloud storage efficiency. Furthermore, we design a reasonable comprehensive efficiency index, in order to achieve the maximum efficiency, and we also find that the Cauchy matrix is more efficient in coding process.

CHAPTER – 9

REFERENCES

- [1] P. Mell and T. Grance, “The NIST definition of cloud computing,” *Nat.Inst. Stand. Technol.*, vol. 53, no. 6, pp. 50–50, 2009.
- [2] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, “A survey of mobile cloud computing: Architecture, applications, and approaches,” *Wireless Commun.Mobile Comput.*, vol. 13, no. 18, pp. 1587–1611, 2013.
- [3] J. Chase, R. Kaewpuang, W. Yonggang, and D. Niyato, “Joint virtual machine and bandwidth allocation in software defined network (sdn) and cloud computing environments,” in *Proc. IEEE Int. Conf. Commun.*, 2014, pp. 2969–2974.
- [4] H. Li, W. Sun, F. Li, and B. Wang, “Secure and privacy-preserving data storage service in public cloud,” *J. Comput. Res. Develop.*, vol. 51, no. 7, pp. 1397–1409, 2014.
- [5] Y. Li, T. Wang, G. Wang, J. Liang, and H. Chen, “Efficient data collection in sensor-cloud system with multiple mobile sinks,” in *Proc. Adv. Serv. Comput., 10th Asia-Pac. Serv. Comput. Conf.*, 2016, pp. 130–143.
- [6] L. Xiao, Q. Li, and J. Liu, “Survey on secure cloud storage,” *J. Data Acquis. Process.*, vol. 31, no. 3, pp. 464–472, 2016.
- [7] R. J. McEliece and D. V. Sarwate, “On sharing secrets and reed-solomon codes,” *Commun. ACM*, vol. 24, no. 9, pp. 583–584, 1981.
- [8] Yang, P., Xiong, N., & Ren, J. (2020). Data security and privacy protection for cloud storage: A survey. *IEEE Access*, 8, 131723-131740.
- [9] Chen, D., & Zhao, H. (2012, March). Data security and privacy protection issues in cloud computing. In *2012 International Conference on Computer Science and Electronics Engineering* (Vol. 1, pp. 647-651). IEEE.
- [10] Sun, Y., Zhang, J., Xiong, Y., & Zhu, G. (2014). Data security and privacy in cloud computing. *International Journal of Distributed Sensor Networks*, 10(7), 190903.
- [11] Yang, P., Xiong, N., & Ren, J. (2020). Data security and privacy protection for cloud storage: A survey. *IEEE Access*, 8, 131723-131740.
- [12] Chen, D., & Zhao, H. (2012, March). Data security and privacy protection issues in cloud computing. In *2012 International Conference on Computer Science and Electronics Engineering* (Vol. 1, pp. 647-651). IEEE.
- [13] Sun, Y., Zhang, J., Xiong, Y., & Zhu, G. (2014). Data security and privacy in cloud computing. *International Journal of Distributed Sensor Networks*, 10(7), 190903.
- [14] Shariati, S. M., & Ahmadzadegan, M. H. (2015, November). Challenges and security issues in cloud computing from two perspectives: Data security and privacy protection. In *2015 2nd International Conference on Knowledge-Based Engineering and Innovation (KBEI)* (pp. 1078-1082). IEEE.
- [15] A. Akavia, S. Goldwasser, and V. Vaikuntanathan, “Simultaneous hardcore bits and cryptography against memory attacks,” in *Proc. TCC, San Francisco, CA, USA*, 2009, pp. 474–495.
- [16] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, “Provable data possession at untrusted stores,” in *Proc. ACM-CSS, New York*,

NY, USA, 2007, pp. 598–609.

[17] N. Attrapadung and H. Imai, “Attribute-based encryption supporting direct/indirect revocation modes,” in Proc. IMACC, Cirencester, U.K., Dec. 2009, pp. 278–330.

[18] J. Baek, R. Safavi-Naini, and W. Susilo, “Public key encryption with keyword search revisited,” in Proc. ICCSA, Perugia, Italy, 2008, pp. 1249–1259.

[19] M. Bellare, A. Boldyreva, and A. O’Neill, “Deterministic and efficiently searchable encryption,” in Proc. CRYPTO, Santa Barbara, CA, USA, 2007, pp. 535–552.

[20] F. Berti, O. Pereira, T. Peters, and F. X. Standaert, “On leakage-resilient authenticated encryption with decryption leakages,” IACR Trans. Symmetric Cryptol., vol. 2017, no. 3, pp. 271–293, 2017.

[21] J. Bethencourt, A. Sahai, and B. Waters, “Ciphertext-policy attributebased encryption,” in Proc. IEEE Symp. Secur. Privacy, Berkeley, CA, USA, May 2007, pp. 321–334.

[22] T. Bhatia and A. K. Verma, “Data security in mobile cloud computing paradigm: A survey, taxonomy and open research issues,” J. Supercomput., vol. 73, no. 6, pp. 2558–2631, Jun. 2017.

[23] A. Boldyreva, V. Goyal, and V. Kumar, “Identity-based encryption with efficient revocation,” in Proc. ACM CCS, Alexandria, VA, USA, 2008, pp. 417–426.

[24] D. Boneh and X. Boyen, “Efficient selective-ID secure identity based encryption without random oracles,” in Proc. Adv. Cryptol. (Eurocrypt), Interlaken, Switzerland, vol. 3027, 2004, pp. 223–238.

[25] D. Boneh and X. Boyen, “Secure identity-based encryption without random oracles,” in Proc. CRYPTO, vol. 3152. Berlin, Germany: Springer, 2004, pp. 443–459.

[26] D. Boneh and M. Franklin, “Identity-based encryption from the Weil pairing,” in Proc. CRYPTO, vol. 2139. Berlin, Germany: Springer, 2001, pp. 213–229.

[27] D. Boneh, G. Di Crescenzo, R. Ostrovsky, and G. Persiano, “Public key encryption with keyword search,” in Proc. EUROCRYPT, vol. 3027. Berlin, Germany: Springer, 2004, pp. 506–522.

[28] Z. Brakerski, “Fully homomorphic encryption without modulus switching from classical GapSVP,” in Proc. CRYPTO, Santa Barbara, CA, USA, 2012, pp. 868–886.

[29] Z. Brakerski, C. Gentry, and V. Vaikuntanathan, “(Leveled) fully homomorphic encryption without bootstrapping,” ACM Trans. Comput. Theory, vol. 6, no. 3, pp. 1–36, Jul. 2014.

[30] Z. Brakerski and V. Vaikuntanathan, “Efficient fully homomorphic encryption from (standard) LWE,” SIAM Journal on Computing, vol. 43, no. 2, pp. 831–871, Jan. 2014.

[31] Z. Brakerski, Y. T. Kalai, J. Katz, and V. Vaikuntanathan, “Overcoming the hole in the bucket: Public-key cryptography resilient to continual memory leakage,” in Proc. IEEE 51st Annu. Symp. Found. Comput. Sci. (FOCS), Las Vegas, NV, USA, Oct. 2010, pp. 501–510.

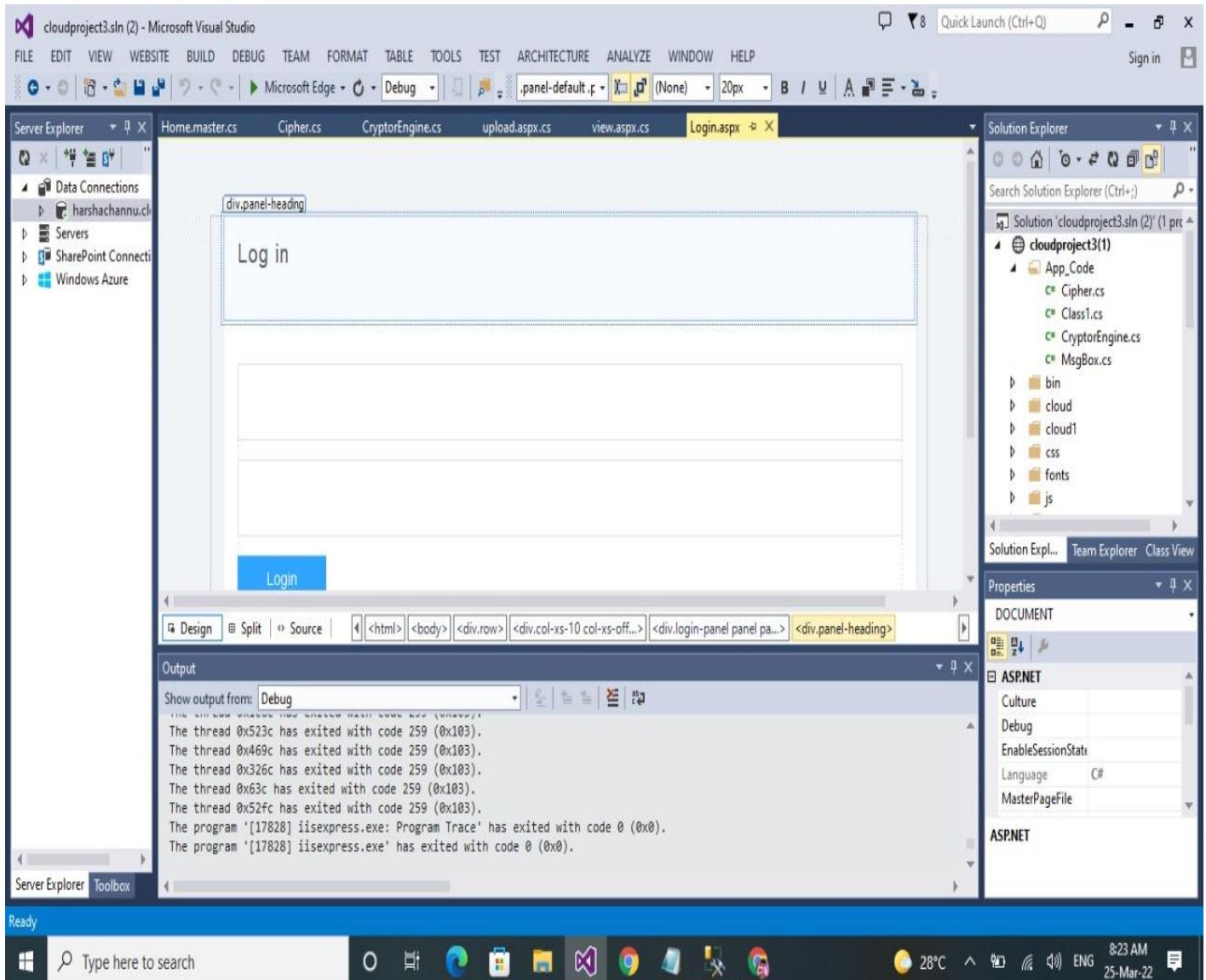
[32] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, “Privacy-preserving multikeyword ranked search over encrypted cloud data,” IEEE Trans. Parallel Distrib. Syst., vol. 25, no. 1, pp. 222–233, Jan. 2014.

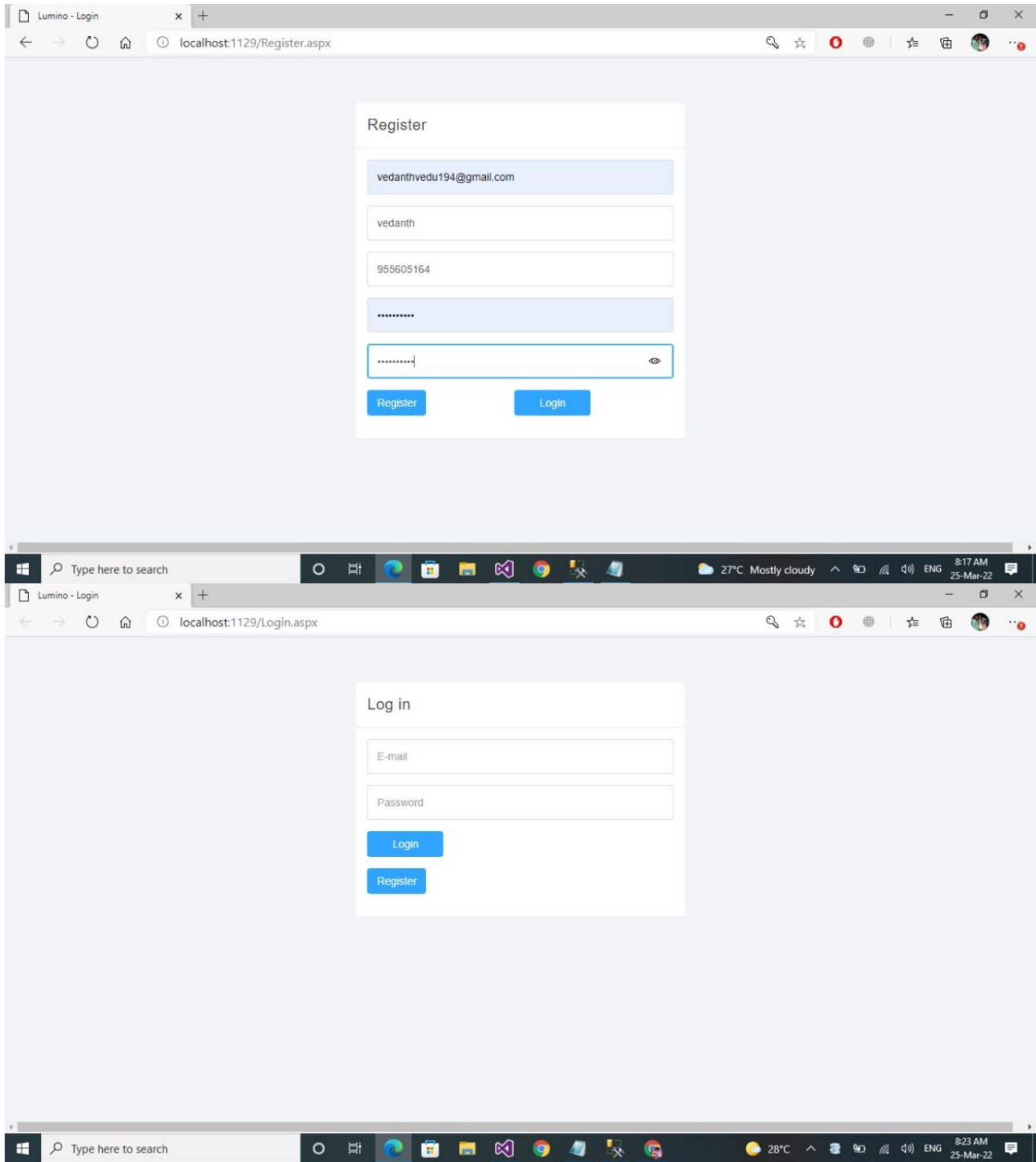
[33] B. Casemore, “Network modernization: Essential for digital transformation and multicloud,” IDC, Framingham, MA, USA, White Paper US45603019, Nov. 2019. [20] J. H. Cheon, A. Kim, M. Kim, and Y. Song, “Homomorphic encryption for arithmetic of approximate numbers,” in Proc. Int. Conf. Theory Appl. Cryptol. Inf. Secur., Hong Kong, 2017, pp. 409–437.

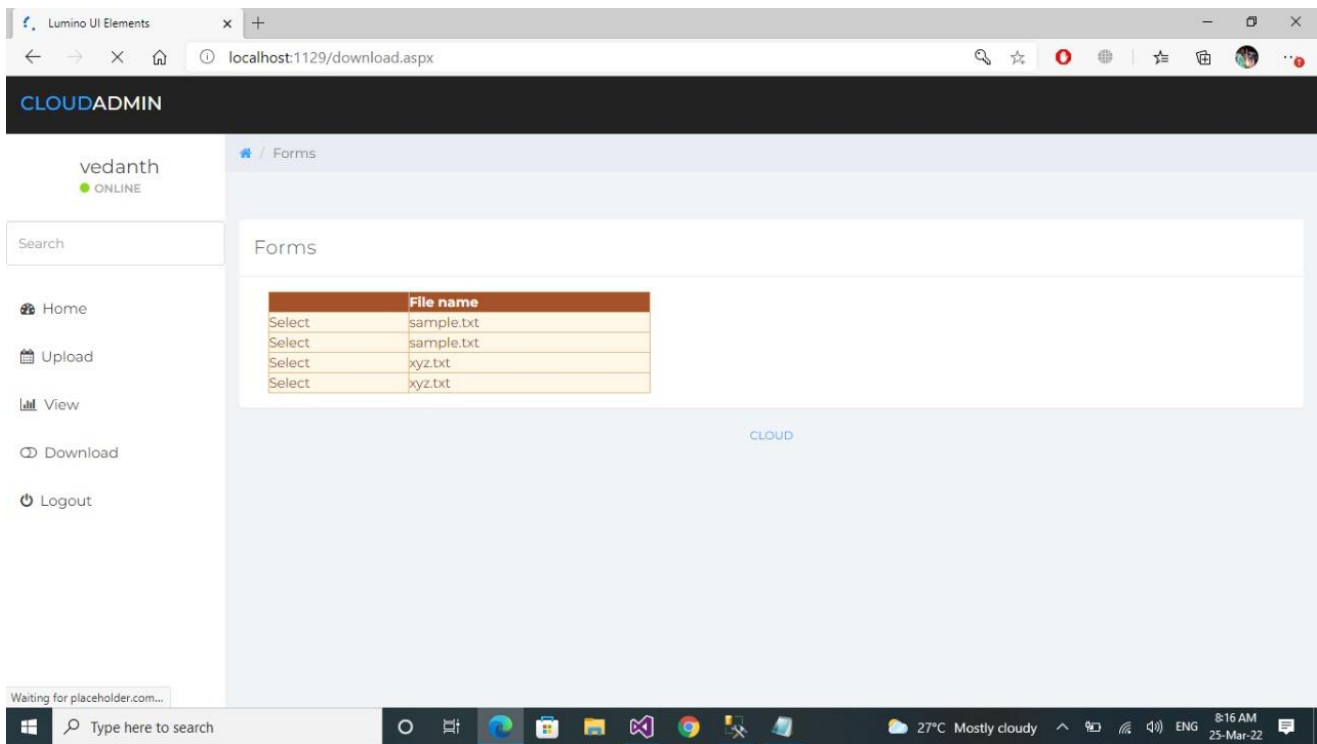
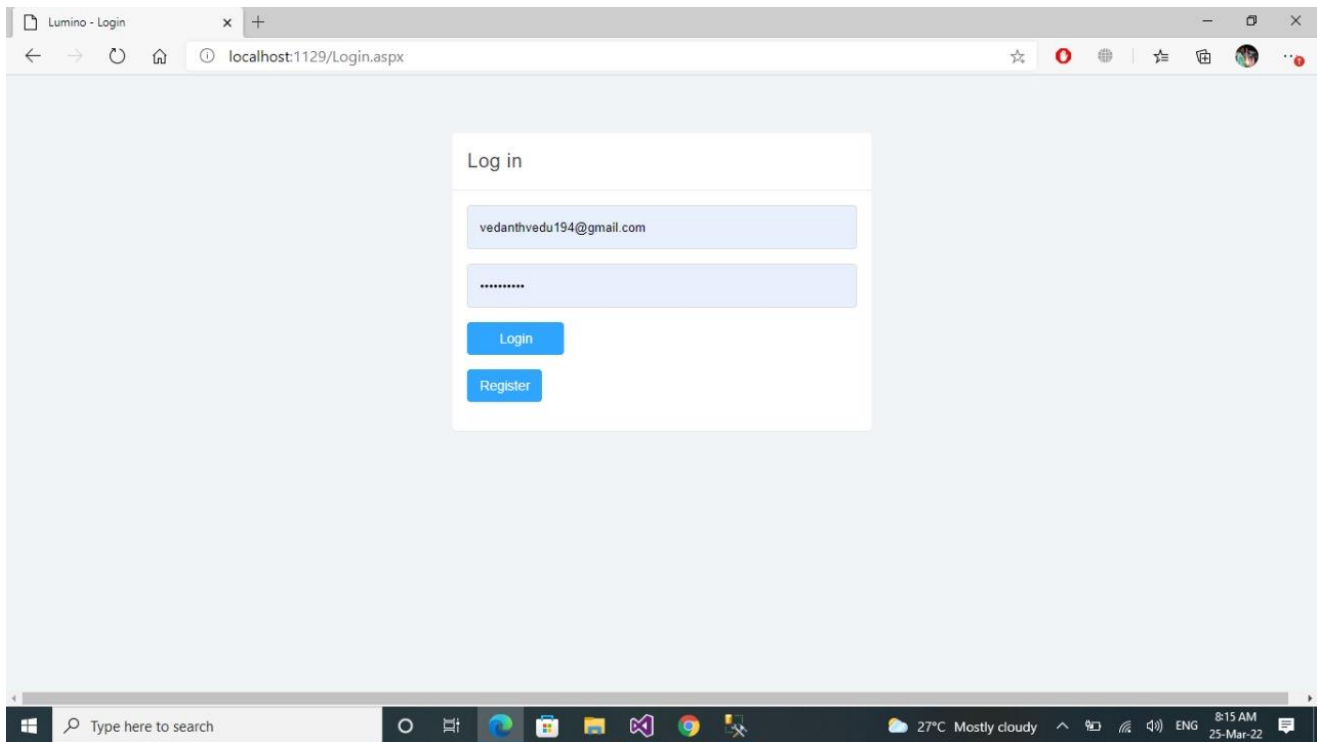
- [34] P. Mell and T. Grance, "The NIST definition of cloud computing," *Nat. Inst. Stand. Technol.*, vol. 53, no. 6, pp. 50–50, 2009.
- [35] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, "A survey of mobile cloud computing: Architecture, applications, and approaches," *Wireless Commun. Mobile Comput.*, vol. 13, no. 18, pp. 1587–1611, 2013.
- [36] J. Chase, R. Kaewpuang, W. Yonggang, and D. Niyato, "Joint virtual machine and bandwidth allocation in software defined network (sdn) and cloud computing environments," in *Proc. IEEE Int. Conf. Commun.*, 2014, pp. 2969–2974.
- [37] H. Li, W. Sun, F. Li, and B. Wang, "Secure and privacy-preserving data storage service in public cloud," *J. Comput. Res. Develop.*, vol. 51, no. 7, pp. 1397–1409, 2014.
- [38] Y. Li, T. Wang, G. Wang, J. Liang, and H. Chen, "Efficient data collection in sensor-cloud system with multiple mobile sinks," in *Proc. Adv. Serv. Comput., 10th Asia-Pac. Serv. Comput. Conf.*, 2016, pp. 130–143.
- [39] L. Xiao, Q. Li, and J. Liu, "Survey on secure cloud storage," *J. Data Acquis. Process.*, vol. 31, no. 3, pp. 464–472, 2016.
- [40] R. J. McEliece and D. V. Sarwate, "On sharing secrets and reed-solomon codes," *Commun. ACM*, vol. 24, no. 9, pp. 583–584, 1981.
- [41] J. S. Plank, "T1: Erasure codes for storage applications," in *Proc. 4th USENIX Conf. File Storage Technol.*, 2005, pp. 1–74.
- [42] R. Kulkarni, A. Forster, and G. Venayagamoorthy, "Computational intelligence in wireless sensor networks: A survey," *IEEE Commun. Surv. Tuts.*, vol. 13, no. 1, pp. 68–96, First Quarter 2011.
- [43] Z. Xia, X. Wang, L. Zhang, Z. Qin, X. Sun, and K. Ren, "A privacy-preserving and copy-deterrence content-based image retrieval scheme in cloud computing," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 11, pp. 2594–2608, Nov. 2016.

CHAPTER - 10

APPENDIX :







CLOUDADMIN

vedanth
ONLINE

Search

Home

Upload

View

Download

Logout

Forms

	File name
Select	sample.txt
Select	sample.txt
Select	xyz.txt
Select	xyz.txt

Cloud Data

Fog Data

System Data

Decrypt

Decrypt

Decrypt

Cloud Data

Fog Data

System Data

Waiting for placeholder.com...

Type here to search

27°C Mostly cloudy

8:16 AM 25-Mar-22

CLOUDADMIN

vedanth
ONLINE

Search

Home

Upload

View

Download

Logout

Forms

	File name
Select	sample.txt
Select	sample.txt
Select	xyz.txt
Select	xyz.txt

Cloud Data

Fog Data

System Data

Decrypt

Decrypt

Decrypt

Cloud Data

Fog Data

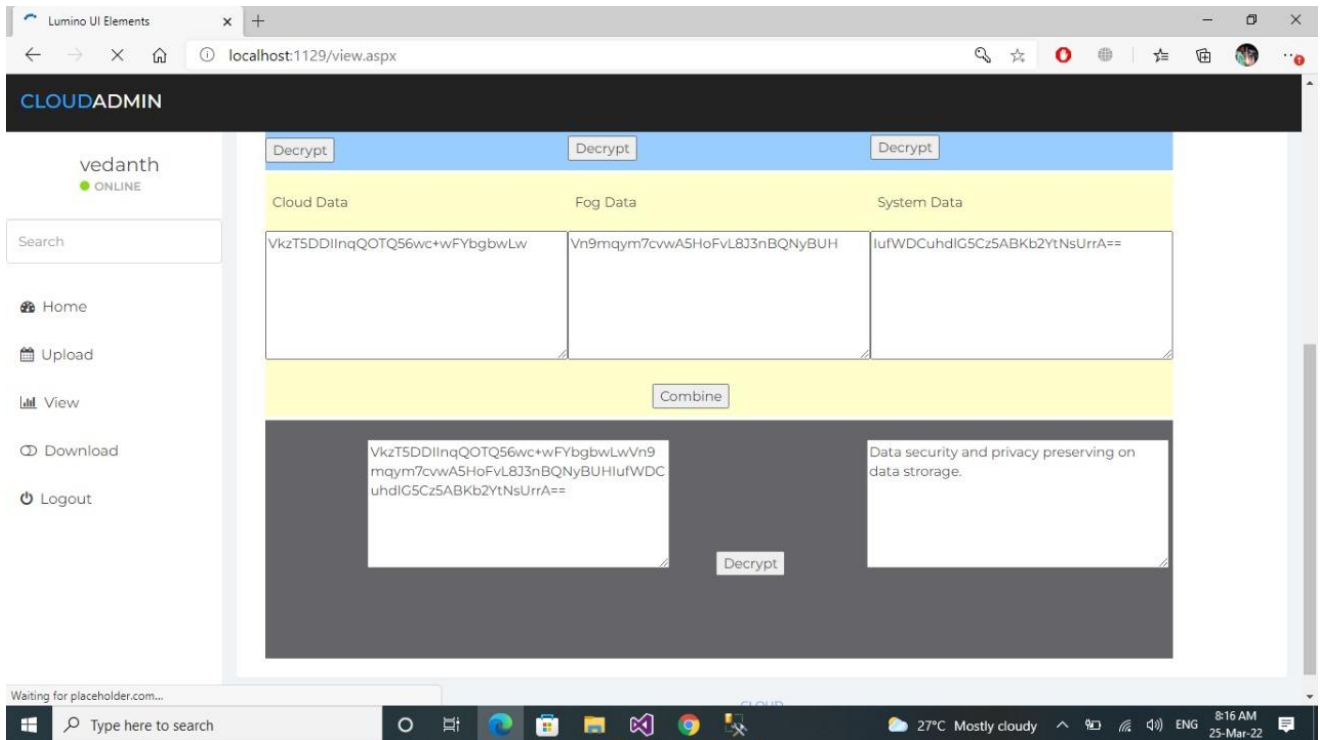
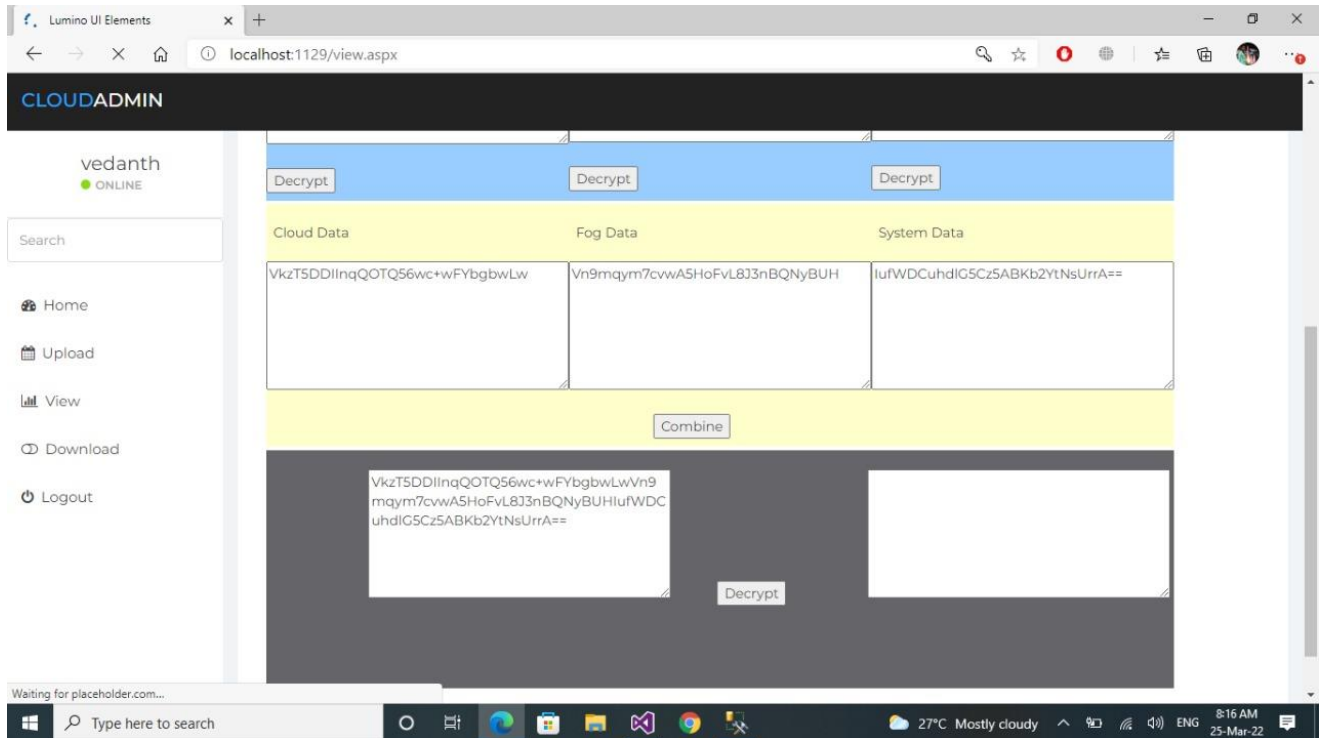
System Data

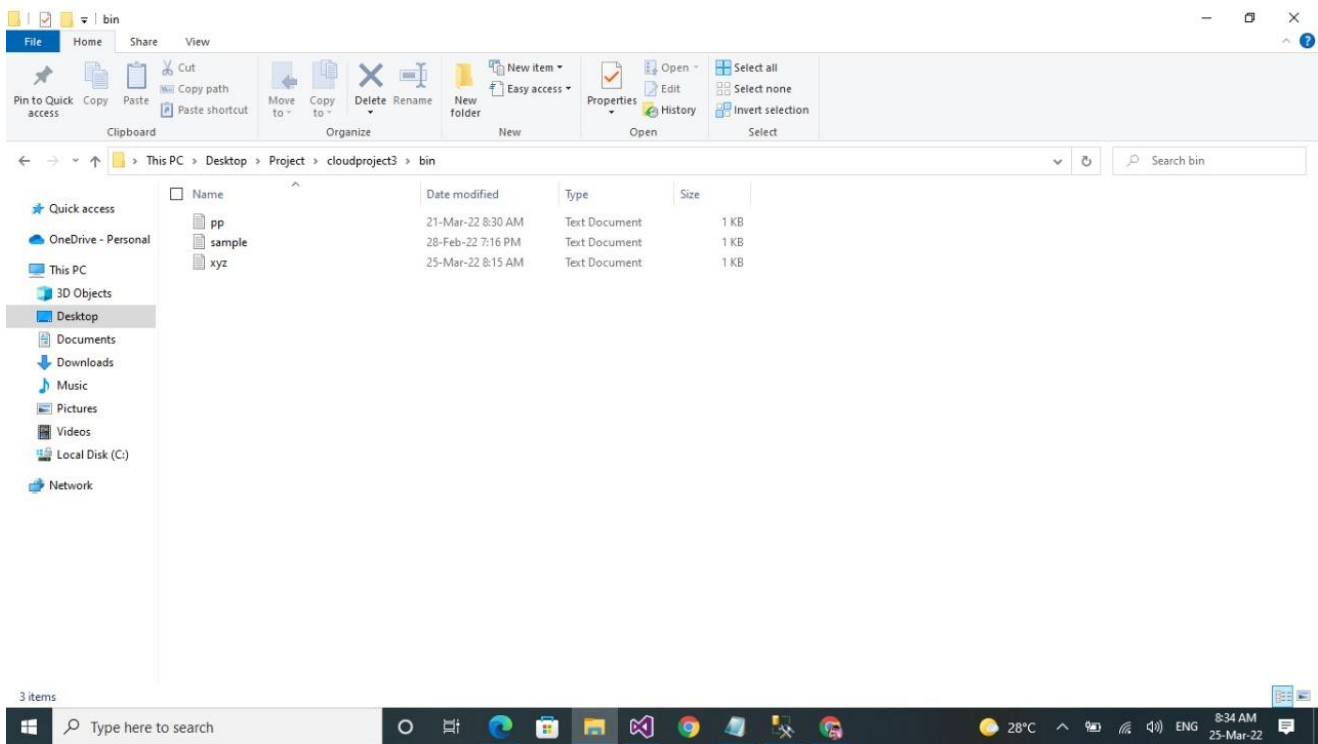
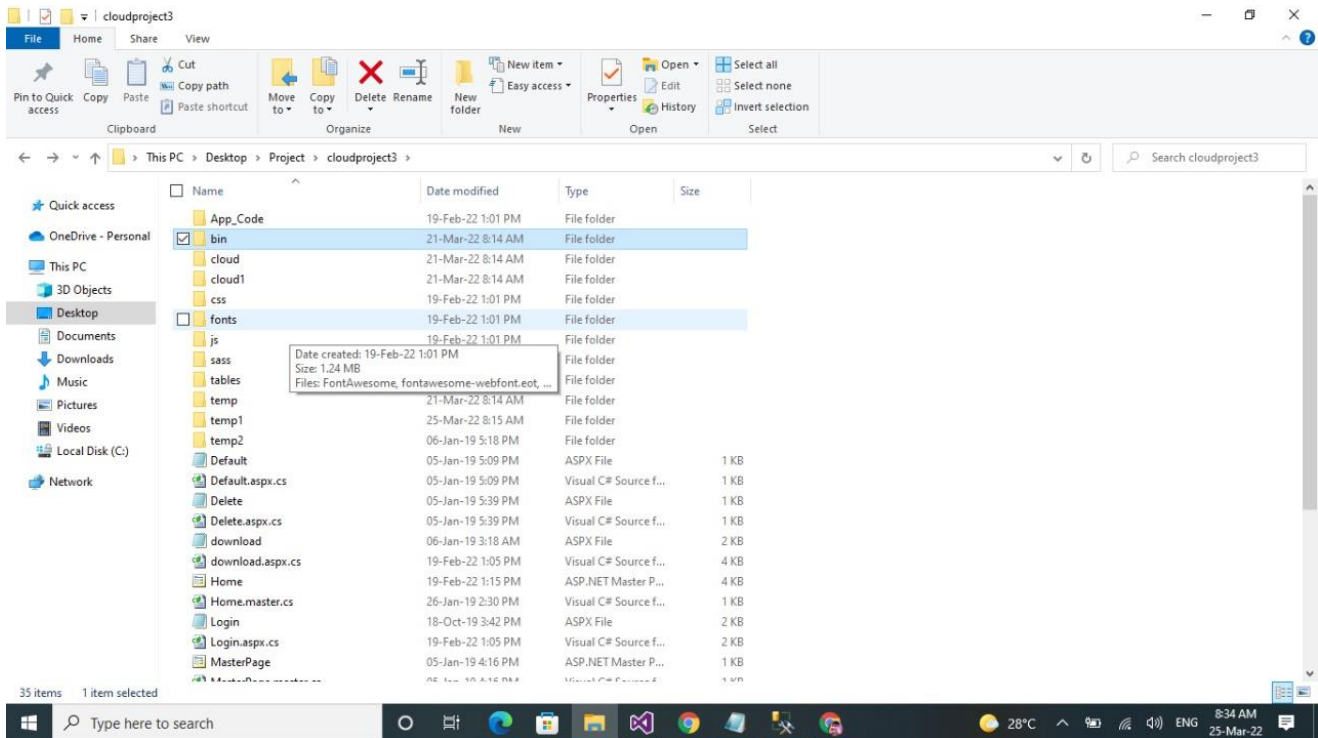
Waiting for placeholder.com...

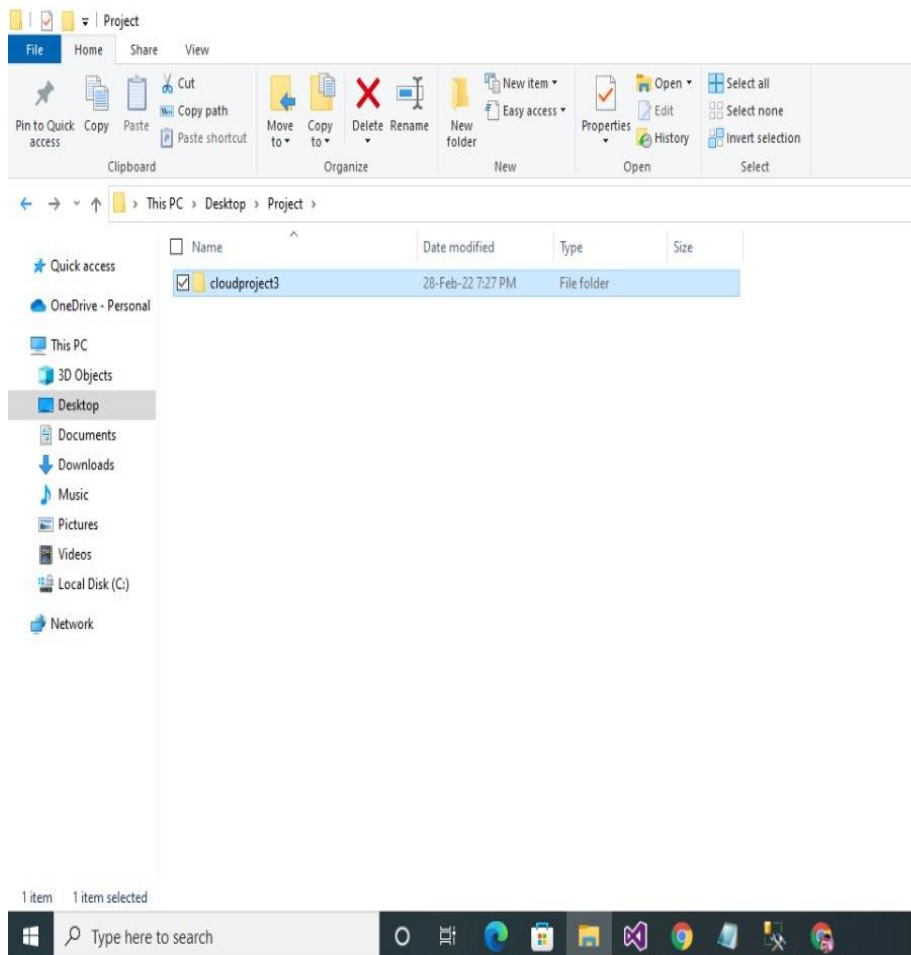
Type here to search

27°C Mostly cloudy

8:16 AM 25-Mar-22







CHAPTER-11

11.1 EXECUTION PROCESS

Backend process:

- ☐ Connect to MySQL database.
- ☐ Go to start → run → SQLyog → ok
- ☐ It will be connected to MySQL database.
- ☐ Create a new user for your project.
- ☐ Sql>create user username identified by password ; → enter
- ☐ Sql>grant dba to username ; → enter
- ☐ Sql>type password ; → enter
- ☐ Sql>select * from tab ; → enter
- ☐ No rows are selected because it is the new user of our project.
- ☐ Then minimize SQLyog.
- ☐ Click on the project code, next click → databases, click → databases.sql.
- ☐ Right click → databases.sql and click → edit on notepad++.
- ☐ The tables will be imported and automatically the imp.exe command prompt closes.
- ☐ Again go to SQLyog.
- ☐ Sql>select * from tab;
- ☐ Tables will be displayed.

Frontend:

Using Net beans

1.Project

Deployment

2.Connectivity

3. Run to execute

1. Project Deployment:

Click on NetBeans IDE ☐ file ☐ import ☐ existing projects into workspace ☐ copy code path

☐ click finish.

2. Connectivity

- ☐ Start NetBeans IDE.
- ☐ In the IDE, choose File > New Project.
- ☐ in the Name and Location page of the wizard,
- ☐ In the Project Name field, type
- ☐ Leave the Use Dedicated Folder for Storing Libraries checkbox unselected.
- ☐ In the Create Main Class field,

You have left the Create Main Class checkbox selected in the New Project wizard, the IDE has created a skeleton main class

- ☐ Save the change by choosing File > Save.

3. Run to execute:

To click Run as ☐ Run on the project

Using Tomcat

Go to project open it -> source code/code -> code for tomcat -> copy the code and paste in web-apps (open tomcat -> web apps) -> code -> web root -> WEB-INF -> config -> system.

Properties -> open it Change duser, dpass as you created for the project.

url=jdbc: oracle: thin: @localhost:1521: xe

In URL we have hostname, port number, service name to change this names and number. Go to oracle where it is installed -> open it and search for tnsnames.ora -> save.

.

Start Tomcat server then go web browser type URL as
<http://localhost:8081>

Select Tomcat manager->enter
Select your project

11.2 SOURCE CODE:

```
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
using System.Security.Cryptography;
using System.Text;
using System.Web;

/// <summary>
/// Summary description for Cipher
/// </summary>
public class Cipher
{
    /// <summary>
    /// Encrypt a string.
    /// </summary>
    /// <param name="plainText">String to be encrypted</param>
    /// <param name="password">Password</param>
    ///
    public static string password = "p@SSword";
    public Cipher()
    {
    }

    public string Encrypt(string plainText, string password)
    {
        if (plainText == null)
        {
            return null;
        }

        if (password == null)
        {
            password = String.Empty;
        }

        // Get the bytes of the string
        var bytesToBeEncrypted = Encoding.UTF8.GetBytes(plainText);
        var passwordBytes = Encoding.UTF8.GetBytes(password);

        // Hash the password with SHA256
        passwordBytes = SHA256.Create().ComputeHash(passwordBytes);

        var bytesEncrypted = Cipher.Encrypt(bytesToBeEncrypted, passwordBytes);
```

```

    return Convert.ToBase64String(bytesEncrypted);
}

/// <summary>
/// Decrypt a string.
/// </summary>
/// <param name="encryptedText">String to be decrypted</param>
/// <param name="password">Password used during encryption</param>
/// <exception cref="FormatException"></exception>
public string Decrypt(string encryptedText, string password)
{
    if (encryptedText == null)
    {
        return null;
    }

    if (password == null)
    {
        password = String.Empty;
    }

    // Get the bytes of the string
    var bytesToBeDecrypted = Convert.FromBase64String(encryptedText);
    var passwordBytes = Encoding.UTF8.GetBytes(password);

    passwordBytes = SHA256.Create().ComputeHash(passwordBytes);

    var bytesDecrypted = Cipher.Decrypt(bytesToBeDecrypted, passwordBytes);

    return Encoding.UTF8.GetString(bytesDecrypted);
}

private static byte[] Encrypt(byte[] bytesToBeEncrypted, byte[] passwordBytes)
{
    byte[] encryptedBytes = null;

    // Set your salt here, change it to meet your flavor:
    // The salt bytes must be at least 8 bytes.
    var saltBytes = new byte[] { 1, 2, 3, 4, 5, 6, 7, 8 };

    using (MemoryStream ms = new MemoryStream())
    {
        using (RijndaelManaged AES = new RijndaelManaged())
        {
            var key = new Rfc2898DeriveBytes(passwordBytes, saltBytes, 1000);

            AES.KeySize = 256;
            AES.BlockSize = 128;

```

```

        AES.Key = key.GetBytes(AES.KeySize / 8);
        AES.IV = key.GetBytes(AES.BlockSize / 8);

        AES.Mode = CipherMode.CBC;

        using (var cs = new CryptoStream(ms, AES.CreateEncryptor(), CryptoStreamMode.Write))
        {
            cs.Write(bytesToBeEncrypted, 0, bytesToBeEncrypted.Length);
            cs.Close();
        }

        encryptedBytes = ms.ToArray();
    }
}

return encryptedBytes;
}

private static byte[] Decrypt(byte[] bytesToBeDecrypted, byte[] passwordBytes)
{
    byte[] decryptedBytes = null;

    // Set your salt here, change it to meet your flavor:
    // The salt bytes must be at least 8 bytes.
    var saltBytes = new byte[] { 1, 2, 3, 4, 5, 6, 7, 8 };

    using (MemoryStream ms = new MemoryStream())
    {
        using (RijndaelManaged AES = new RijndaelManaged())
        {
            var key = new Rfc2898DeriveBytes(passwordBytes, saltBytes, 1000);

            AES.KeySize = 256;
            AES.BlockSize = 128;
            AES.Key = key.GetBytes(AES.KeySize / 8);
            AES.IV = key.GetBytes(AES.BlockSize / 8);
            AES.Mode = CipherMode.CBC;

            using (var cs = new CryptoStream(ms, AES.CreateDecryptor(), CryptoStreamMode.Write))
            {
                cs.Write(bytesToBeDecrypted, 0, bytesToBeDecrypted.Length);
                cs.Close();
            }

            decryptedBytes = ms.ToArray();
        }
    }
}

```

```

        return decryptedBytes;
    }
}

```

```

using System;
using System.Collections.Generic;
using System.Configuration;
using System.Linq;
using System.Security.Cryptography;
using System.Text;
using System.Web;

/// <summary>
/// Summary description for CryptorEngine
/// </summary>
public class CryptorEngine
{
    /// <summary>
    /// Encrypt a string using dual encryption method. Return a encrypted cipher Text
    /// </summary>
    /// <param name="toEncrypt">string to be encrypted</param>
    /// <param name="useHashing">use hashing? send to for extra security</param>
    /// <returns></returns>
    public static string Encrypt(string toEncrypt, bool useHashing)
    {
        byte[] keyArray;
        byte[] toEncryptArray = UTF8Encoding.UTF8.GetBytes(toEncrypt);

        System.Configuration.AppSettingsReader settingsReader = new AppSettingsReader();
        // Get the key from config file
        string key = "Syed Moshiur Murshed";
        //System.Windows.Forms.MessageBox.Show(key);
        if (useHashing)
        {
            MD5CryptoServiceProvider hashmd5 = new MD5CryptoServiceProvider();
            keyArray = hashmd5.ComputeHash(UTF8Encoding.UTF8.GetBytes(key));
            hashmd5.Clear();
        }
        else
            keyArray = UTF8Encoding.UTF8.GetBytes(key);

        TripleDESCryptoServiceProvider tdes = new TripleDESCryptoServiceProvider();
        tdes.Key = keyArray;

```

```

    tdes.Mode = CipherMode.ECB;
    tdes.Padding = PaddingMode.PKCS7;

    ICryptoTransform cTransform = tdes.CreateEncryptor();
    byte[] resultArray = cTransform.TransformFinalBlock(toEncryptArray, 0, toEncryptArray.Length);
    tdes.Clear();
    return Convert.ToBase64String(resultArray, 0, resultArray.Length);
}
/// <summary>
/// DeCrypt a string using dual encryption method. Return a DeCrypted clear string
/// </summary>
/// <param name="cipherString">encrypted string</param>
/// <param name="useHashing">Did you use hashing to encrypt this data? pass true is yes</param>
/// <returns></returns>
public static string Decrypt(string cipherString, bool useHashing)
{
    byte[] keyArray;
    byte[] toEncryptArray = Convert.FromBase64String(cipherString);

    System.Configuration.AppSettingsReader settingsReader = new AppSettingsReader();
    //Get your key from config file to open the lock!
    string key = "Syed Moshir Murshed";

    if (useHashing)
    {
        MD5CryptoServiceProvider hashmd5 = new MD5CryptoServiceProvider();
        keyArray = hashmd5.ComputeHash(UTF8Encoding.UTF8.GetBytes(key));
        hashmd5.Clear();
    }
    else
        keyArray = UTF8Encoding.UTF8.GetBytes(key);

    TripleDESCryptoServiceProvider tdes = new TripleDESCryptoServiceProvider();
    tdes.Key = keyArray;
    tdes.Mode = CipherMode.ECB;
    tdes.Padding = PaddingMode.PKCS7;

    ICryptoTransform cTransform = tdes.CreateDecryptor();
    byte[] resultArray = cTransform.TransformFinalBlock(toEncryptArray, 0, toEncryptArray.Length);

    tdes.Clear();
    return UTF8Encoding.UTF8.GetString(resultArray);
}
}

```

LOGIN CODE:

```
using System;
using System.Collections.Generic;
using System.Data.SqlClient;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class Login : System.Web.UI.Page
{
    SqlConnection con;
    SqlDataReader rs;
    string conn = "Data Source=HARSHACHANNU;Initial
Catalog=cloud1;Integrated Security=True";
    protected void Page_Load(object sender, EventArgs e)
    {

    }
    protected void Button2_Click(object sender, EventArgs e)
    {
        Response.Redirect("Register.aspx");
    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        con = new SqlConnection(conn);
        con.Open();
        string s1;
        s1 = "select * from login1 where mailid='" + TextBox1.Text + "' and pwd='" +
        TextBox2.Text + "'";
        SqlCommand cmd = new SqlCommand(s1, con);
        rs = cmd.ExecuteReader();
        if (rs.Read())
        {

            string id = Convert.ToString(rs.GetValue(1));
            Session["uname"] = id.ToString();
            Session["mailid"] = TextBox1.Text;
```

```

        Response.Redirect("Default.aspx");
        rs.Close();
    }
    else
    {
        MsgBox.Show("Invalid username or password");
        TextBox1.Text = "";
        TextBox2.Text = "";
        rs.Close();
    }
}

```

REGISTER CODE:

```

using System;
using System.Collections.Generic;
using System.Data.SqlClient;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class Register : System.Web.UI.Page
{
    SqlConnection con;
    SqlDataReader rs;
    string conn = "Data Source=HARSHACHANNU;Initial
Catalog=cloud1;Integrated Security=True";
    protected void Page_Load(object sender, EventArgs e)
    {

    }

    protected void Button2_Click(object sender, EventArgs e)
    {
        if (TextBox1.Text == "" || TextBox2.Text == "" || TextBox1.Text == "" ||
            TextBox2.Text == "" || TextBox2.Text == "")

```

```

    {
        MsgBox.Show("Fill All Datas");
    }
    else if (TextBox2.Text != TextBox5.Text)
    {
        MsgBox.Show("Password not matching");
    }
    else
    {
        con = new SqlConnection(conn);
        string s1 = "insert into login1 values('" + TextBox1.Text + "','" +
        TextBox3.Text + "','" + TextBox4.Text + "','" + TextBox2.Text + "')";
        con.Open();
        SqlCommand cmd = new SqlCommand(s1, con);
        cmd.ExecuteNonQuery();
        TextBox1.Text = "";
        TextBox2.Text = "";
        TextBox3.Text = "";
        TextBox4.Text = "";
        TextBox5.Text = "";
        Response.Redirect("Login.aspx");
    }
}
protected void Button1_Click(object sender, EventArgs e)
{
    Response.Redirect("Login.aspx");
}
}

```

UPLOAD CODE:

```

using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
using System.Security.Cryptography;
using System.Text;
using System.Web;
using System.Web.UI;

```

```

using System.Web.UI.WebControls;
using System.Data.SqlClient;

public partial class upload : System.Web.UI.Page
{
    public static string dt;
    public static string conn = "Data Source=HARSHACHANNU;Initial
Catalog=cloud1;Integrated Security=True";
    SqlConnection con;
    SqlCommand cmd;
    SqlDataReader rs;
    protected void Page_Load(object sender, EventArgs e)
    {

    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        if (FileUpload1.HasFile)
        {
            try
            {
                if (FileUpload1.PostedFile.ContentType == "text/plain")
                {
                    if (FileUpload1.PostedFile.ContentLength < 102400)
                    {
                        string filename = Path.GetFileName(FileUpload1.FileName);
                        FileUpload1.SaveAs(Server.MapPath("~/temp1/") + filename);
                        var fileStream = new FileStream(Server.MapPath("~/temp1/") +
filename, FileMode.Open, FileAccess.Read);
                        using (var streamReader = new StreamReader(fileStream,
Encoding.UTF8))
                        {
                            dt = streamReader.ReadToEnd();
                        }
                        string path = Server.MapPath("~/temp1/") + filename;
                        FileInfo file = new FileInfo(path);
                        if (file.Exists)//check file exsist or not
                        {

```

```

        file.Delete();

    }
    Cipher c = new Cipher();
    string c_text= Convert.ToString( c.Encrypt(dt, "p@SSword"));
    int a = c_text.Length;
    int x =(int) a / 3;
    string sub = c_text.Substring(0, x);
    string sub1 = c_text.Substring(x, x);
    int x1 = a-(x + x);
    string sub2 = c_text.Substring(x+x, x1);
    string cipherText1 = CryptorEngine.Encrypt(sub, true);
    string cipherText2 = CryptorEngine.Encrypt(sub1, true);
    string cipherText3 = CryptorEngine.Encrypt(sub2, true);
    File.WriteAllText(Server.MapPath("~/cloud/") + filename,
String.Empty);
        using (StreamWriter writer = new
StreamWriter(Server.MapPath("~/cloud/") + filename))
        {
            writer.Write(cipherText1);

        }
        File.WriteAllText(Server.MapPath("~/cloud1/") + filename,
String.Empty);
        using (StreamWriter writer = new
StreamWriter(Server.MapPath("~/cloud1/") + filename))
        {
            writer.Write(cipherText2);

        }
        File.WriteAllText(Server.MapPath("~/temp/") + filename,
String.Empty);
        using (StreamWriter writer = new
StreamWriter(Server.MapPath("~/temp/") + filename))
        {
            writer.Write(cipherText3);

        }
        File.WriteAllText(Server.MapPath("~/bin/") + filename,

```

```

using System.Data.SqlClient;
using System.IO;
using System.Linq;
using System.Text;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class view : System.Web.UI.Page
{
    public static int xa = 0;
    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack)
        {
            if (string.IsNullOrEmpty(Session["mailid"] as string))
            {
                Response.Redirect("Login.aspx");
            }
            else
            {
                SqlConnection con = new SqlConnection("Data
Source=HARSHACHANNU;Initial Catalog=cloud1;Integrated Security=True");
                string strSQL = "Select * from file_m where uname=" +
Session["mailid"].ToString() + """;
                SqlDataAdapter dt = new SqlDataAdapter(strSQL, con);
                DataSet ds = new DataSet();
                dt.Fill(ds, "uname");
                con.Close();
                GridView1.DataSource = ds;
                GridView1.DataBind();
            }
        }

        protected void GridView1_SelectedIndexChanged(object sender, EventArgs e)
        {
            TextBox1.Text = "";

```

```

        TextBox4.Text = "";
        TextBox5.Text = "";
        TextBox6.Text = "";
        TextBox7.Text = "";
        TextBox8.Text = "";
        TextBox9.Text = "";
        TextBox10.Text = "";
        string f_name = Convert.ToString(GridView1.SelectedRow.Cells[1].Text);
        string dt, dt1, dt2;
        var fileStream = new FileStream(Server.MapPath("~/cloud/" + f_name),
        FileMode.Open, FileAccess.Read);
        using (var streamReader = new StreamReader(fileStream,
        Encoding.UTF8))
        {
            dt = streamReader.ReadToEnd();
        }
        var fileStream1 = new FileStream(Server.MapPath("~/cloud1/" + f_name),
        FileMode.Open, FileAccess.Read);
        using (var streamReader = new StreamReader(fileStream1,
        Encoding.UTF8))
        {
            dt1 = streamReader.ReadToEnd();
        }
        string path1 = Server.MapPath("~/temp/" + f_name);
        FileInfo file1 = new FileInfo(path1);
        if (file1.Exists)//check file exists or not
        {
            var fileStream2 = new FileStream(Server.MapPath("~/temp/" + f_name),
            FileMode.Open, FileAccess.Read);
            using (var streamReader = new StreamReader(fileStream2,
            Encoding.UTF8))
            {
                dt2 = streamReader.ReadToEnd();
            }
        }
        else
        {

```

```

        var fileStream2 = new FileStream(Server.MapPath("~/bin/" + f_name),
        FileMode.Open, FileAccess.Read);
        using (var streamReader = new StreamReader(fileStream2,
        Encoding.UTF8))
        {
            dt2 = streamReader.ReadToEnd();
        }
    }
    TextBox1.Text = dt;
    TextBox4.Text = dt1;
    TextBox5.Text = dt2;

```

```

}
protected void Button4_Click(object sender, EventArgs e)
{
    string plain1 = CryptorEngine.Decrypt(TextBox1.Text, true);
    TextBox6.Text = plain1;
    xa = xa + 1;
    if(xa==3)
    {
        xa = 0;
        Button6.Visible = true;
    }
}

```

```

protected void Button2_Click(object sender, EventArgs e)
{
    string plain1 = CryptorEngine.Decrypt(TextBox4.Text, true);
    TextBox7.Text = plain1;
    xa = xa + 1;
    if (xa == 3)
    {
        xa = 0;
        Button6.Visible = true;
    }
}

```

```

protected void Button3_Click(object sender, EventArgs e)
{

```

```

    string plain1 = CryptorEngine.Decrypt(TextBox5.Text, true);
    TextBox8.Text = plain1;
    xa = xa + 1;
    if (xa == 3)
    {
        xa = 0;
        Button6.Visible = true;
    }
}
protected void Button6_Click(object sender, EventArgs e)
{
    string txt = TextBox6.Text + TextBox7.Text + TextBox8.Text;

    TextBox9.Text = txt;
}
protected void Button5_Click(object sender, EventArgs e)
{
    Cipher c = new Cipher();
    string p_text = Convert.ToString(c.Decrypt(TextBox9.Text, "p@SSword"));
    TextBox10.Text = p_text;
}
}

```