**SUMMER TRAINING REPORT**

**On**

**Development of Secure Data**

**Storage Drive**

**Submitted by**

**Deeksha kashyap**

**171500091**

Department of Computer Engineering & Applications

**Institute of Engineering & Technology**



**GLA University**

**Mathura- 281406, INDIA**

**2019**

**Department of computer Engineering and Applications**

**GLA University, Mathura**

**17 km. Stone NH#2, Mathura-Delhi Road, P.O. – Chaumuha,**

**Mathura – 281406**



**Declaration**

I hereby declare that the work which is being presented in the Summer Training “**Development of Secure Data Storage Drive”,** in partial fulfillment of the requirements for Summer Training viva voce, is an authentic record of my own work carried under the supervision of “**MICRANTHROPE ADVANCED SYSTEM & TECHNOLOGIES PVT.LTD**” talking to ADRDE, Defence Research & Development Organization, Ministry of Defence, GOI Establishment, Agra**.**

**Signature of Candidate:**

**Name of Candidate: Deeksha kashyap**

**Roll. No. : 171500091**

**Course: B.tech (Computer Science and Engineering)**

**Year: 3rd**

**Semester: 5th**

**Synopsis**

**Student Information:**

|  |  |
| --- | --- |
| Name: Deeksha kashyap | University Roll. No.171500091 |
| Mobile: 9412254060 | Email:diksha.gla\_cs17@gla.ac.in |

**Information about Industry/Organization:**

|  |  |
| --- | --- |
| Industry/Organization Name with full Address | Micranthrope Advanced Systems & Technologies Pvt.Ltd.  19B, V.S.,Palace, Sector 4, Avas Vikas, Bodla, Agra-282007 |
| Contact Person | Name & Designation: R.N Singh  Mobile/email: www.micranthrope.com(9411205230) |

**Project Information:**

|  |  |
| --- | --- |
| Title Of Project/Training/Task | Development of Secure Data Storage Drive |
| Role & Responsibility |  |
| Technical Details | Software Requirements: Visual Studio(.Net) |
| Training Implementation Details | Fully Implemented |
| Training Period | Start Date:15 May 2019  End Date:14 June 2019  Duration Of Training (In Weeks):6 weeks |

**Summary of the Training Work:**

|  |
| --- |
| *Third-party cloud services* are provided for a fee by an external organization, such as Google, Microsoft Corporation or Amazon.com. Examples of cloud-storage applications include Dropbox, Microsoft OneDrive and Google Drive. Examples of cloud-based application services include Onai and Common Care. *Two-factor or multifactor authentication* requires a user to provide two or more pieces of authenticating evidence before receiving access: generally, something the user knows (e.g., a password) plus something the user has (e.g., an access code). Multifactor authentication may include a biometric, such as a fingerprint. |

**ACKNOWLEDGEMENT**

The success and final outcome of this project required a lot of guidance and assistance from many people and I am extremely privileged to have got this all along the completion of my project. All that I have done is only due to such supervision and assistance and I would not forget to thank them.

I respect and thank **Mr. Ashutosh Singh**, for providing me an opportunity to do the project work in company **Micranthrope Advanced System & Technologies Pvt. Ltd.** and giving me all support and guidance which made me complete the project duly. I am extremely thankful to him for providing such a nice support and guidance, although he had busy schedule managing the corporate affairs.

I own my deep gratitude to our project guide **Mr. Ashutosh Singh** who took keen interest on my project work and guided me all along, till the completion of our project work by providing all the necessary information for developing a good project.

I would like to thank the staff for the operation extended to us throughout the project. After doing this project I can confidently say that this experience has not only enriched me with technical knowledge but also has unparsed the maturity of thought and vision. The attributes required in being a successful professional.

Deeksha kashyap

**Abstract**

The secure USB flash drive was developed to improve the security of the conventional USB flash drive, which is vulnerable to leakages of internally stored data caused by extortion, loss, etc. However, it has been continuously reported that the secure USB flash drive, which protects data through the adoption of a wide range of security technologies in wide-ranging ways, cannot assure data security because of implementation and environmental vulnerabilities, eavesdropping, unlock commands, and reverse engineering. As such, there is growing demand for a more powerful secure USB flash drive to solve these fundamental problems. Therefore, this paper presents a secure USB mechanism that prevents leakages of authentication data and does not compare authentication data for smart human care services, which have been a fundamental problem of existing flash drives. The proposed mechanism provides better security than the existing secure USB flash drive by satisfying the need for confidentiality, integrity, authentication, and access control and safely protecting data from impersonation, man-in-the-middle, replay, and eavesdropping attacks by malicious attackers. An assessment of its security using the formalized verification tool VISUAL STUDIO has proved that it is safe. Therefore, it is considered that a safer, more secure USB flash drive can be manufactured using the mechanism proposed in this paper.

**Contents**

Certificate

Acknowledgments

Abstract

1. **Introduction**  1

1.1 Purpose………………………………………………………………………….2

1.2 Motivation………………………………………………………………………2

**2. Security Services 3**

2.1 Authentication and Authorization………………………………………………3

2.2Confidentiality…………………………………………………………………..3

2.3Key Management………………………………………………………………..3

2.4 Performance……………………………………………………………………..4

**3. General Management**   **5**

3.1 Quality System………………………………………………………………….5

3.1.1 ISO 9001…………………………………………………………………..5

3.1.2 Six Sigma………………………………………………………………….7

3.1.3 SEI CMM………………………………………………………………….8

**4. Technical Management 10**

4.1 Software Engineering Model……………………………………………………10

4.1.1 XP………………………………………………………………………….10

4.1.2 ASD………………………………………………………………………..11

4.1.3 DSDM……………………………………………………………………..12

**5. Database Tables**  **14**

**6. Program 15**

**DEVELOPMENT OF SECURE DATA STORAGE DRIVE**

**INTRODUCTION**

USB storage devices (flash drives, USB sticks etc.) offer many advantages for us. However, at the same time, they cause security problems because it is easy to copy a lot of files to a tiny USB memory in a few seconds. We might have some secure data on our PC which we do not want other users to copy through the USB. Therefore, we many need to define a USB storage policy to make USB drives write protected or not to be accessed through the system.

The USB flash drive is mostly used for data storage and backup, booting disks, and portable program storage, and it must be capable of storing many different types of data in order to provide such functions. The various types of stored data include sensitive data such as public certificate, confidential business data, and personal information. Therefore, a USB flash drive with guaranteed security is essential since serious damage can occur if such information is extorted by a malicious attacker.

Despite positive purposes, USB flash drives have problems regarding sensitive data leakage due to no additional security functions. Such a problem is the root cause of security threats that expose sensitive data through extortion and loss, and the reason for the threat is the fact that the data in the flash drive are stored in a raw form, rather than being encrypted in an altered form to cope with external attacks. Security threats by extortion enable a malicious attacker to extort a victim’s private data stored inside a USB flash drive by seizing and inserting them into his own computer. There is a more serious problem in that there can be secondary or tertiary victims if the extorted data are items of personal information such as public certificate or confidential business information. The security threat by loss refers to cases which a flash drive is lost as a result of a victim’s error. As with the security threat by extortion, a third party who obtains a lost drive can take the data stored inside the drive by inserting it into his or her computer.

* 1. **Purpose**

Secure data storage applies to data at rest stored in computer/server hard disks, portable devices – like external hard drives or USB drives – as well as online/cloud, network-based storage area network (SAN) or network attached storage (NAS) systems.

Secure data storage is achieved in the following ways:

* Data encryption
* Access control mechanism at each data storage device/software
* Protection against viruses, worms and other data corruption threats
* Physical/manned storage device and infrastructure security
* Enforcement and implementation of layered/tiered storage security architecture

Secure data storage is essential for organizations which deal with sensitive data, both in order to avoid data theft, as well as to ensure uninterrupted operations.

* 1. **Motivation**

Due to wide variety of available security solutions and different techniques to arrive at a particular solution, it is really important to review the entire field prior to selecting a particular approach that satisfies the requirement. This survey will provide an overview of prominent characteristics of the existing storage security solutions to help make an informed decision. Developing and understanding of these aspects of secure storage will help to motivate future research in this area. A major problem associated with securing storage is to properly weigh the tradeoffs between costs and benefits associated with the security measure. In this survey I will discuss about important security issues associated with different types of storage and will present a comprehensive overview of those security approaches mainly using confidentiality, integrity, authenticity and availability as a framework.

**SECURITY SERVICES**

This section will give an overview of some of the basic set of services that every storage system designer should consider.

**2.1** **Authentication and Authorization**

Authentication is the process of corroborating the identity of a particular principle or the source of the message. Message authentication is performed by an entity to authenticate the origin of the message, claimed to be from another entity. Authentication can be mutual, both the server and the client must prove their identity to the other party. The purpose of granting appropriate privileges to the entity, established via authentication is called authorization.

**2.2** **Confidentiality**

Securing data while in transit as well as when it is stored in the physical media is crucial. Confidentiality can be achieved by using appropriate encryption schemes. Data can be secured while in traffic using protocols such as IPsec and SSL. In this case, servers decrypt the data before storing it on the media, using temporary session keys negotiated at the start of the session. End-to-end security is achieved by writers themselves encrypting the data before sending it to the storage server and readers decrypting the data at the host.

**2.3 Key Management**

End-to-end security places minimal trust on the servers and the data is accessible only to users with appropriate keys. Efficient and scalable management and distribution of these keys is of paramount important, as revoking and adding a user to the group or merging two groups can require complete reencryption or re-distribution. Another important aspect of key management is the key recovery. Further care has to be taken for storing, archiving and deleting these keys.

**2.4 Performance**

The two goals of a storage system, efficient storage and security, intrinsically conflict with each other. Each additional security measure requires some extra computation that will detract the system’s ability to perform other storage related computation. Usability and Manageability also plays an important role in a storage system’s overall impact.

**GENERAL MANAGEMENT**

**3.1 Quality System**

* **ISO 9001**

ISO 9001 is defined as the international standard that specifies requirements for a [quality management system (QMS)](https://asq.org/quality-resources/quality-management-system). Organizations use the standard to demonstrate the ability to consistently provide products and services that meet customer and regulatory requirements. It is the most popular standard in the [ISO 9000 series](https://asq.org/quality-resources/iso-9000) and the only standard in the series to which organizations can certify.

ISO 9001 was first published in 1987 by the [International Organization for Standardization (ISO)](https://asq.org/quality-resources/standards-101#iso), an international agency composed of the national standards bodies of more than 160 countries. The [current version of ISO 9001](https://asq.org/quality-press/display-item?item=T1040) was released in **September 2015.**

### What topics does ISO 9001:2015 cover?

ISO 9001 is based on the [plan-do-check-act methodology](https://asq.org/quality-resources/pdca-cycle) and provides a [process-oriented approach](https://asq.org/quality-progress/2001/12/standards-outlook/the-process-approach-to-qms-in-iso-9001-and.html) to documenting and reviewing the structure, responsibilities, and procedures required to achieve effective quality management in an organization. Specific sections of the standard contain information on many topics, such as:

* Requirements for a [quality management system](https://asq.org/quality-resources/quality-management-system), including documented information, planning and determining process interactions
* Responsibilities of management
* Management of resources, including human resources and an organization’s work environment
* Product realization, including the steps from design to delivery
* Measurement, analysis, and improvement of the QMS through activities like [internal audits](https://asq.org/quality-resources/auditing) and corrective and preventive action

Changes introduced in the 2015 ISO 9001 revision are intended to ensure that ISO 9001 continues to adapt to the changing environments in which organizations operate. Some of the key updates in ISO 9001:2015 include:

* The introduction of new terminology
* Restructuring some of the information
* An emphasis on risk-based thinking to enhance the application of the process approach
* Improved applicability for services
* Increased leadership requirements



**WHAT ARE THE BENEFITS OF ISO 9001?**

ISO 9001 helps organizations ensure their customers consistently receive high quality products and services, which in turn brings many benefits, including [satisfied customers](https://asq.org/quality-resources/customer-satisfaction), management, and employees.

Because ISO 9001 specifies the requirements for an effective quality management system, organizations find that using the standard helps them:

* Organize a [quality management system (QMS)](https://asq.org/quality-resources/quality-management-system)
* Create satisfied customers, management, and employees
* Continually improve their processes
* Save costs

In Nevada, the [Clark County School District used ISO 9001](https://secure.asq.org/perl/msg.pl?prvurl=http://asq.org/2006/05/iso-9000/nevada-school-district-saves-millions-through-process-management.pdf) to save $174 million over 10 years in actual expenditures and cost avoidance. More than 3,000 employees were trained to the standard, enabling three critical components of the system’s success: training, communication and respect, and efficiency.

* **SIX SIGMA**

Six Sigma is a set of management tools and techniques designed to improve business by reducing the likelihood of error. It is a data-driven approach which uses statistical methodology for eliminating defects.

The etymology is based on the Greek symbol “sigma” or "σ”, a statistical term for measuring process deviation from the process mean or target. “Six Sigma” comes from the bell curve used in statistics, where one Sigma symbolizes a single standard deviation from the mean. If the process has six Sigmas, three above and three below the mean, the defect rate is classified as “extremely low.”

* **The 5 Key Principles of Six Sigma**

1. **Focus on the customer**   
   This is based on the popular belief that the “customer is the king.” The primary goal is to bring maximum benefit to the customer. For this, a business needs to understand its customers, their needs, and what drives sales or loyalty. This requires establishing the standard of quality as defined by what the customer or market demands.
2. **Measure the value stream and find your problem**   
   Map the steps in a given process to determine areas of waste. Gather data to discover the specific problem area that is to be addressed or transformed. Have clearly defined goals for data collection, including defining the data to be collected, the reason for the data gathering, insights expected, ensuring the accuracy of measurements, and establishing a standardized data collection system. Ascertain if the data is helping to achieve the goals, whether or not the data needs to be refined, or additional information collected. Identify the problem. Ask questions and find the root cause.
3. **Get rid of the junk**   
   Once the problem is identified, make changes to the process to eliminate variation, thus removing defects. Remove the activities in the process that do not add to the customer value. If the value stream doesn’t reveal where the problem lies, tools are used to help discover the outliers and problem areas. Streamline functions to achieve quality control and efficiency. In the end, by taking out the above mentioned junk, bottlenecks in the process are removed.
4. **Keep the ball rolling**Involve all stakeholders. Adopt a structured process where your team contributes and collaborates their varied expertise for the purposes of problem-solving.  
   Six Sigma processes can have a great impact on an organization, so the team has to be proficient in the principles and methodologies used. Hence, specialized training and knowledge are required to reduce the risk of project or re-design failures, and ensure that the process performs optimally.
5. **Ensure a flexible and responsive ecosystem**   
   The essence of Six Sigma is business transformation and change. When a faulty or inefficient process is removed, it calls for a change in the work practice and employee approach. A robust culture of flexibility and responsiveness to changes in procedures can ensure a streamlined project implementation. The people and departments involved should be able to adapt to change with ease, so in order to facilitate this, processes should be designed for quick and seamless adoption. Ultimately, the company that has an eye fixed on the data, examines the bottom line periodically, and adjusts its processes where necessary, can gain a competitive edge.

* **SEI CMM**

**SEI** refers to Software Engineering Institute of Carnegie Mellon University in Pittsburgh, it helps to improve software development process. SEI offers technical guidance for advanced practice in software engineering.

**CMM** refers to ‘Capability Maturity Model’, presently called as CMMI ‘Capability Maturity Model Integration’; it was developed by SEI and used to understand the processes followed in as organization. CMMI ratings were given to organizations after review by auditors.

The Capability Maturity Model (CMM) is a methodology used to develop and refine an organization's software development process. The model describes a five-level evolutionary path of increasingly organized and systematically more mature processes. CMM was developed and is promoted by the [Software Engineering Institute (SEI)](https://searchsoftwarequality.techtarget.com/definition/Software-Engineering-Institute-SEI), a research and development center sponsored by the U.S. Department of Defense (DoD). SEI was founded in 1984 to address software engineering issues and, in a broad sense, to advance software engineering methodologies. More specifically, SEI was established to optimize the process of developing, acquiring, and maintaining heavily software-reliant systems for the DoD. Because the processes involved are equally applicable to the software industry as a whole, SEI advocates industry-wide adoption of the CMM.

CMMI has five levels of process maturity for quality software delivery, they are,

* + Level 1 – In this level only few processes were used and to rely on individual efforts for successful completion of projects.
  + Level 2 – In this level there will be realistic planning like project tracking, requirements analysis, configuration management.
  + Level 3 – In this level a proper software development and maintenance are incorporated throughout the organization. Training programs are exercised for better understanding of the process.
  + Level 4 – In this level various metrics are used to follow the process and products. The delivery quality is high.
  + Level 5 – In this level new process and technology were used and implemented efficiently. The objective will be for consistence process development.

**TECHNICAL MANAGEMENT**

**4.1 Software Engineering Model**

* **XP(EXTREME PROGRAMMING)**

Extreme Programming (XP) is an agile software development framework that aims to produce higher quality software, and higher quality of life for the development team. XP is the most specific of the agile frameworks regarding appropriate engineering practices for software development.

**Basic principles of Extreme programming:**

* **Coding:** The concept of coding which is used in XP model is slightly different from traditional coding. Here, coding activity includes drawing diagrams (modeling) that will be transformed into code, scripting a web-based system and choosing among several alternative solutions.
* **Testing:** XP model gives high importance on testing and considers it be the primary factor to develop fault-free software.
* **Listening:** The developers need to carefully listen to the customers if they have to develop good quality software. Sometimes programmers may not have the depth knowledge of the system to be developed. So, it is desirable for the programmers to understand properly the functionality of the system and they have to listen to the customers.
* **Designing:** Without a proper design, a system implementation becomes too complex and very difficult to understand the solution, thus it makes maintenance expensive. A good design results elimination of complex dependencies within a system. So, effective use of suitable design is emphasized.
* **Feedback:** One of the most important aspects of the XP model is to gain feedback to understand the exact customer needs. Frequent contact with the customer makes the development effective.
* **Simplicity:** The main principle of the XP model is to develop a simple system that will work efficiently in present time, rather than trying to build something that would take time and it may never be used. It focuses on some specific features that are immediately needed, rather than engaging time and effort on speculations of future requirements.

**Applications of Extreme Programming (XP):**

 Some of the projects that are suitable to develop using XP model are given below:

* **Small projects:** XP model is very useful in small projects consisting of small teams as face to face meeting is easier to achieve.
* **Projects involving new technology or Research projects:** This type of projects faces changing of requirements rapidly and technical problems. So XP model is used to complete this type of projects.
* **ASD (ADAPTIVE SOFTWARE DEVELOPMENT)**

 Adaptive Software Development (ASD) is a direct outgrowth of an earlier [agile framework](https://www.productplan.com/glossary/agile-framework/), Rapid Application Development (RAD). It aims to enable teams to quickly and effectively adapt to changing requirements or market needs by evolving their products with lightweight planning and continuous learning. The ASD approach encourages teams to develop according to a three-phase process: speculate, collaborate, and learn.

## Strengths and Weakness of ASD

### ASD’s strengths include:

* Focused on the end users, which can lead to better and more intuitive products
* Allows for on-time and even early delivery
* Encourages more transparency between developers and clients

### ASD’s weaknesses include:

* Demands extensive user involvement, which can be difficult to facilitate
* Integrates testing into every stage, which can add to a project’s costs
* Emphasis on rapid iterating and continuous feedback can lead to scope creep
* **DSDM (DYNAMIC SYSTEM DEVELOPMENT METHOD)**

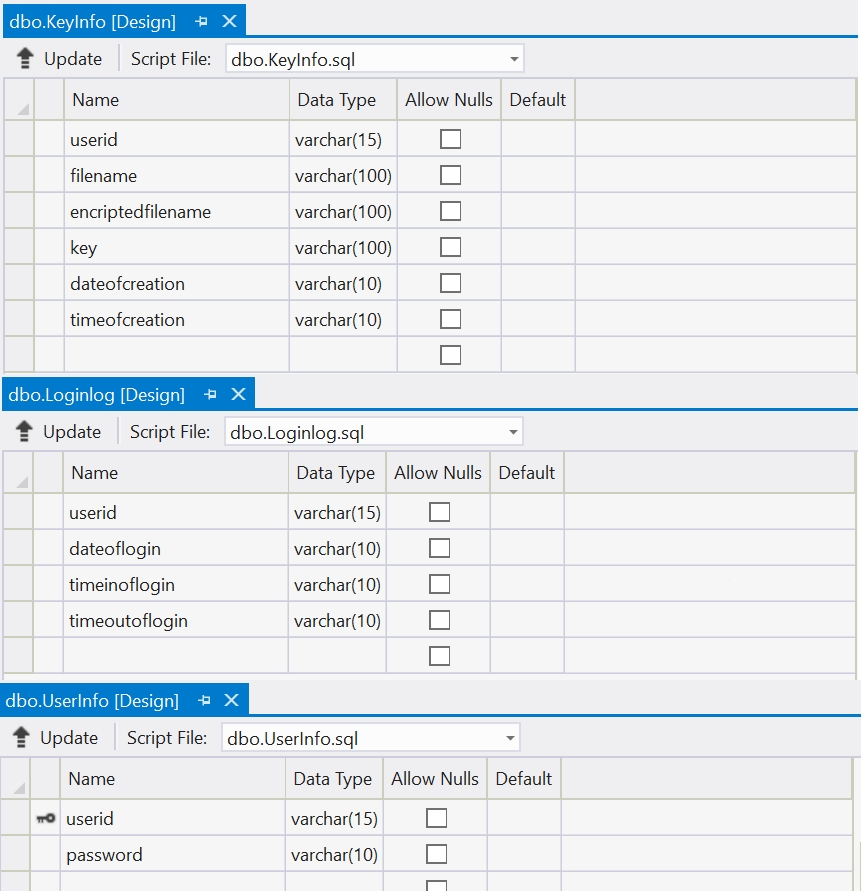
The **Dynamic Systems Development technique (DSDM)** is an associate degree agile code development approach that provides a framework for building and maintaining systems. The DSDM philosophy is borrowed from a modified version of the sociologist principle—80 % of an application is often delivered in twenty percent of the time it’d desire deliver the entire (100 percent) application.

DSDM is an iterative code method within which every iteration follows the 80% rule that simply enough work is needed for every increment to facilitate movement to the following increment. The remaining detail is often completed later once a lot of business necessities are noted or changes are requested and accommodated.

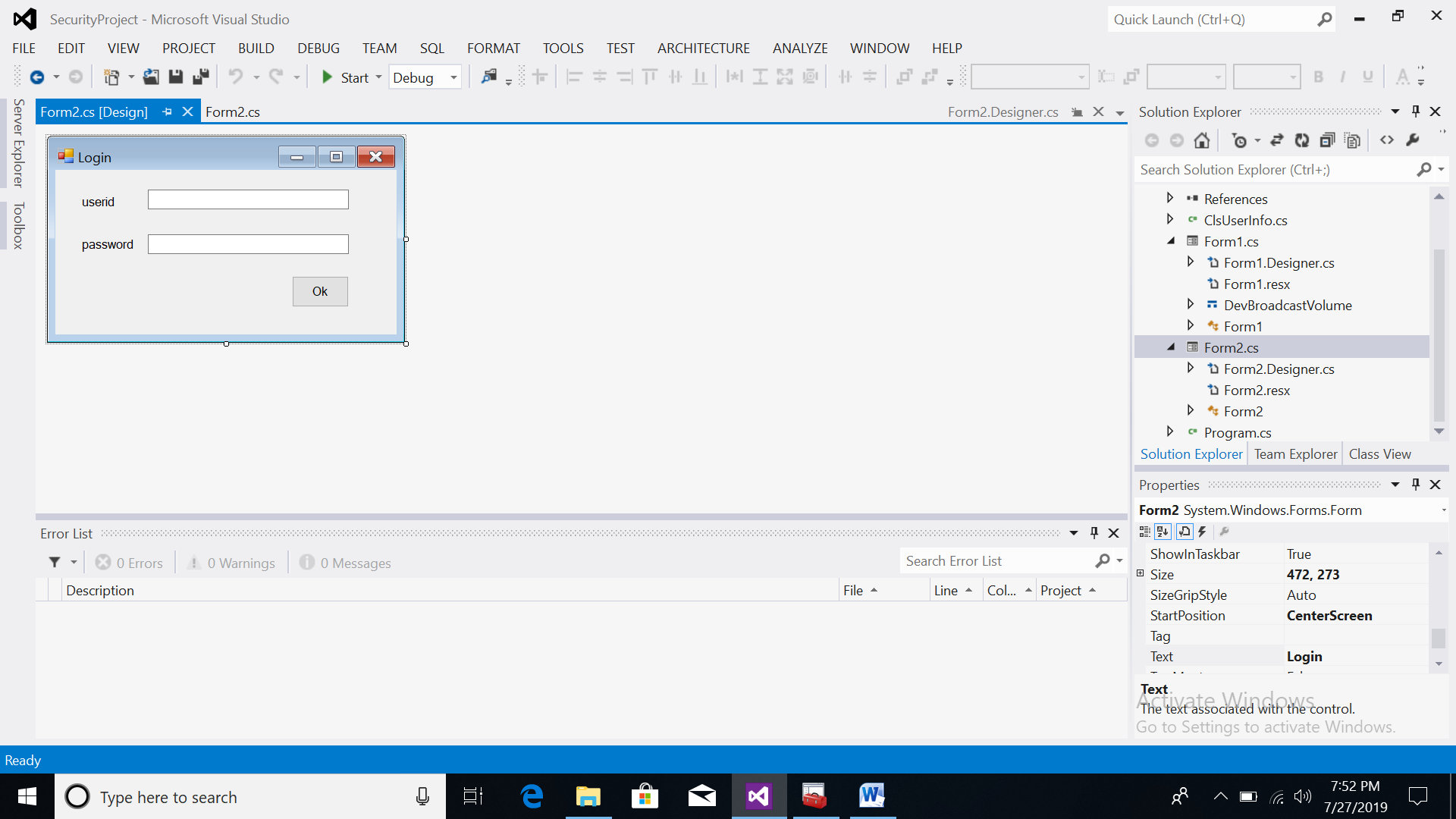
The DSDM tool (www.dsdm.org) could be a worldwide cluster of member companies that put together tackle the role of “keeper” of the strategy. The pool has outlined AN [Agile Development Model](https://www.geeksforgeeks.org/software-engineering-agile-development-models/), known as the DSDM life cycle that defines 3 different unvarying cycles, preceded by 2 further life cycle activities:

1. **Feasibility Steps:**   
   It establishes the essential business necessities and constraints related to the applying to be designed then assesses whether or not the application could be a viable candidate for the DSDM method.
2. **Business Steps:**   
   It establishes the use and knowledge necessities that may permit the applying to supply business value; additionally, it is the essential application design and identifies the maintainability necessities for the applying.
3. **Functional Model Iteration:**   
   It produces a collection of progressive prototypes that demonstrate practicality for the client.  
   (Note: All DSDM prototypes are supposed to evolve into the deliverable application.) The intent throughout this unvarying cycle is to collect further necessities by eliciting feedback from users as they exercise the paradigm.
4. **Design and Build Iteration:**   
   It revisits prototypes designed throughout useful model iteration to make sure that everyone has been designed during a manner that may alter it to supply operational business price for finish users. In some cases, useful model iteration and style and build iteration occur at the same time.
5. **Implementation:**   
   It places the newest code increment (an “operationalized” prototype) into the operational surroundings. It ought to be noted that:
   * **(a)** the increment might not 100% complete or,
   * **(b)** Changes are also requested because the increment is placed into place. In either case, DSDM development work continues by returning to the useful model iteration activity.

**DATABASE TABLES**



**FORM2.DESIGN.CS**



**FORM2.CS Program**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace SecurityProject

{

public partial class Form2 : Form

{

string getDate(DateTime dt)

{

string CurrentDate = dt.Day.ToString() + "/" + dt.Month.ToString() + "/" + dt.Year.ToString();

return CurrentDate;

}

string getTime(DateTime t)

{

string CurrentTime = t.ToShortTimeString();

return CurrentTime;

}

public Form2()

{

InitializeComponent();

Start.ActiveForm.Visible = false;

}

private void label1\_Click(object sender, EventArgs e)

{

}

private void Form2\_Load(object sender, EventArgs e)

{

}

private void btnOk\_Click(object sender, EventArgs e)

{

SqlConnection con = new SqlConnection("Data Source=desktop-kr9upqb;Initial Catalog=dataprotector;Integrated Security=True;Pooling=False");

con.Open();

string query = "select \* from userinfo where userid='" + txtId.Text + "' and password='" + txtPassword.Text + "'";

SqlCommand cmd = new SqlCommand(query, con);

SqlDataReader dr = cmd.ExecuteReader();

if (dr.Read())

{}

if (dr.HasRows)

{

dr.Close();

ClsUserInfo.userid = txtId.Text;

string dateofLogin = getDate(DateTime.Now);

string timeinofLogin = getTime(DateTime.Now);

ClsUserInfo.dateoflogin = dateofLogin;

ClsUserInfo.timeinoflogin = timeinofLogin;

query = "insert into Loginlog values('" + txtId.Text + "','" + dateofLogin + "','" + timeinofLogin + "','\*\*\*')";

cmd = new SqlCommand(query, con);

cmd.ExecuteNonQuery();

Form1 frm = new Form1();

frm.Show();

con.Close();

this.Visible = false;

}

else

{

MessageBox.Show("Wrong UserId/Password ");

txtId.Text = "";

txtPassword.Text = "";

txtId.Focus();

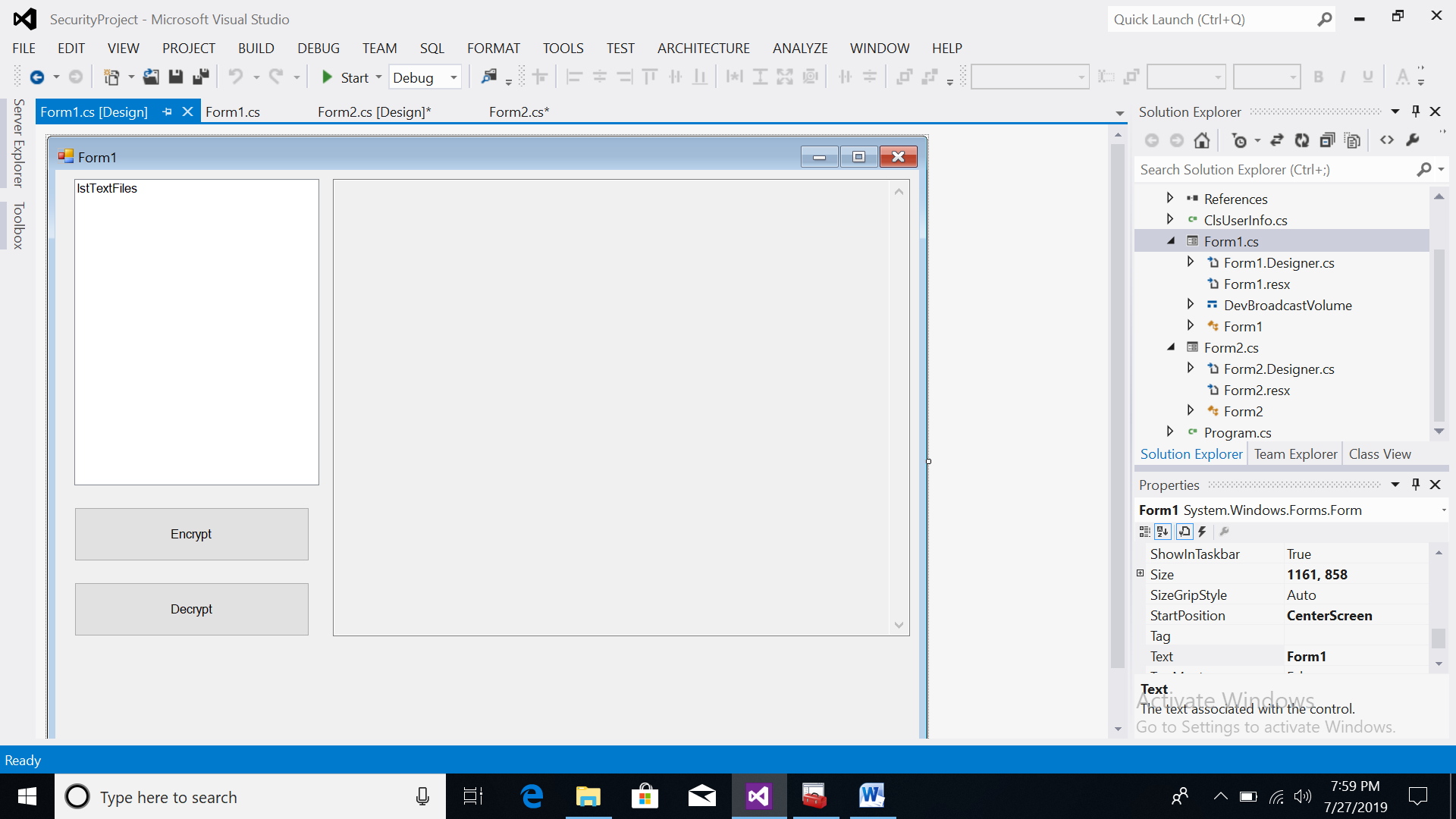
}

}

}

}

**FORM1.DESIGN.CS**



**FORM1.CS Program**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

using System.Data.SqlClient;

using System.IO;

using System.Security;

using System.Security.Cryptography;

using System.Runtime.InteropServices;

namespace SecurityProject

{

[StructLayout(LayoutKind.Sequential)]

public struct DevBroadcastVolume

{

public int Size;

public int DeviceType;

public int Reserved;

public int Mask;

public Int16 Flags;

}

public partial class Form1 : Form

{

private const int WM\_DEVICECHANGE = 0x219;

private const int DBT\_DEVICEARRIVAL = 0x8000;

private const int DBT\_DEVICEREMOVECOMPLETE = 0x8004;

private const int DBT\_DEVTYP\_VOLUME = 0x00000002;

string FileName = "";

string path = "";

protected override void WndProc(ref Message m)

{

base.WndProc(ref m);

switch (m.Msg)

{

case WM\_DEVICECHANGE:

switch ((int)m.WParam)

{

case DBT\_DEVICEARRIVAL:

lstTextFiles.Items.Add("New Device Arrived");

int devType = Marshal.ReadInt32(m.LParam, 4);

DriveInfo[] allDrives = DriveInfo.GetDrives();

foreach (DriveInfo d in allDrives)

{

if (d.DriveType == DriveType.Removable)

{

if (d.VolumeLabel == "SSONI PEN 1")

{

path = d.Name;

FileInfo[] AllTextFiles = GetAllTextFiles(d.Name);

for (int i = 0; i < AllTextFiles.Length; i++)

{

if (AllTextFiles[i].FullName.EndsWith(".txt"))

{

lstTextFiles.Items.Add(AllTextFiles[i].FullName);

}

}

}

}

}

break;

case DBT\_DEVICEREMOVECOMPLETE:

lstTextFiles.Items.Clear();

break;

}

break;

}

}

string getDate(DateTime dt)

{

string CurrentDate = dt.Day.ToString() + "/" + dt.Month.ToString() + "/" + dt.Year.ToString();

return CurrentDate;

}

string getTime(DateTime t)

{

string CurrentTime = t.ToShortTimeString();

return CurrentTime;

}

public Form1()

{

InitializeComponent();

}

FileInfo[] GetAllTextFiles(string path)

{

DirectoryInfo dir = new DirectoryInfo(path);

return dir.GetFiles();

}

[System.Runtime.InteropServices.DllImport("KERNEL32.DLL", EntryPoint = "RtlZeroMemory")]

public static extern bool ZeroMemory(IntPtr Destination, int Length);

// Function to Generate a 64 bits Key.

static string GenerateKey()

{

// Create an instance of Symetric Algorithm. Key and IV is generated automatically.

DESCryptoServiceProvider desCrypto = (DESCryptoServiceProvider)DESCryptoServiceProvider.Create();

// Use the Automatically generated key for Encryption.

return ASCIIEncoding.ASCII.GetString(desCrypto.Key);

}

static void EncryptFile(string sInputFilename, string sOutputFilename, string sKey)

{

try

{

FileStream fsInput = new FileStream(sInputFilename, FileMode.Open, FileAccess.Read);

FileStream fsEncrypted = new FileStream(sOutputFilename, FileMode.Create, FileAccess.Write);

DESCryptoServiceProvider DES = new DESCryptoServiceProvider();

DES.Key = ASCIIEncoding.ASCII.GetBytes(sKey);

DES.IV = ASCIIEncoding.ASCII.GetBytes(sKey);

ICryptoTransform desencrypt = DES.CreateEncryptor();

CryptoStream cryptostream = new CryptoStream(fsEncrypted, desencrypt, CryptoStreamMode.Write);

byte[] bytearrayinput = new byte[fsInput.Length];

fsInput.Read(bytearrayinput, 0, bytearrayinput.Length);

cryptostream.Write(bytearrayinput, 0, bytearrayinput.Length);

cryptostream.Close();

fsInput.Close();

fsEncrypted.Close();

}

catch (Exception e) { MessageBox.Show(e.Message); }

}

public void DecryptFile(string sInputFilename,string sKey)

{

DESCryptoServiceProvider DES = new DESCryptoServiceProvider();

DES.Key = ASCIIEncoding.ASCII.GetBytes(sKey);

DES.IV = ASCIIEncoding.ASCII.GetBytes(sKey);

FileStream fsread = new FileStream(sInputFilename, FileMode.Open, FileAccess.Read);

ICryptoTransform desdecrypt = DES.CreateDecryptor();

CryptoStream cryptostreamDecr = new CryptoStream(fsread, desdecrypt, CryptoStreamMode.Read);

txtDecryptedText.Text = (new StreamReader(cryptostreamDecr).ReadToEnd());

}

void SaveKey(string KeyFileName, string Key)

{

FileStream f = new FileStream(KeyFileName, FileMode.Create);

for (int i = 0; i < Key.Length; i++) f.WriteByte((byte)Key[i]);

}

private void btnSave\_Click(object sender, EventArgs e)

{

if (FileName.Length > 0)

{

SqlConnection con = new SqlConnection("Data Source=desktop-kr9upqb;Initial Catalog=dataprotector;Integrated Security=True;Pooling=False");

con.Open();

string userid = ClsUserInfo.userid;

FileInfo fn = new FileInfo(FileName);

string fName = fn.Name;

Random rnd = new Random();

string EncryptedFileName = fName.Substring(0,fName.IndexOf('.')) + "\_encript.txt";

string KeyFileName = fName.Substring(0, fName.IndexOf('.')) + "\_Key.txt";

string key = GenerateKey();

SaveKey(path+KeyFileName, key);

FileInfo finfo = new FileInfo(KeyFileName);

string dateofcreation = getDate(DateTime.Now);

string timeofcreation = getTime(DateTime.Now);

MessageBox.Show(FileName + "," + path + EncryptedFileName + "," + path + KeyFileName);

EncryptFile(FileName, path+EncryptedFileName, key);

fn = new FileInfo(EncryptedFileName);

EncryptedFileName = fn.Name;

fn = new FileInfo(KeyFileName);

KeyFileName = fn.Name;

string query = "insert into KeyInfo values('" + userid + "','" + fName + "','" + EncryptedFileName + "','" + KeyFileName + "','" + dateofcreation + "','" + timeofcreation + "')";

SqlCommand cmd = new SqlCommand(query, con);

cmd.ExecuteNonQuery();

con.Close();

}

else

{

MessageBox.Show("Please Select the file first ");

}

}

private void lstTextFiles\_Click(object sender, EventArgs e)

{

FileName = lstTextFiles.Items[lstTextFiles.SelectedIndex].ToString();

}

private void btnDecrypt\_Click(object sender, EventArgs e)

{

if (FileName.Length > 0)

{

SqlConnection con = new SqlConnection("Data Source=desktop-kr9upqb;Initial Catalog=dataprotector;Integrated Security=True;Pooling=False");

con.Open();

string fname = FileName;

FileInfo fn = new FileInfo(fname);

fname = fn.Name;

string query = "select \* from keyinfo where userid='" + ClsUserInfo.userid + "' and encriptedfilename='" + fname + "'";

SqlCommand cmd = new SqlCommand(query, con);

MessageBox.Show(query);

SqlDataReader dr = cmd.ExecuteReader();

string key="";

if (dr.Read())

{ }

if (dr.HasRows && path.Length>0)

{

key=dr[3].ToString();

dr.Close();

StreamReader sm = new StreamReader(path+key);

String KeyText = sm.ReadToEnd();

MessageBox.Show((path+fname+":")+(path + key + ":") + KeyText);

DecryptFile(path+fname, KeyText);

MessageBox.Show(KeyText);

}

con.Close();

}

}

private void Form1\_FormClosed(object sender, FormClosedEventArgs e)

{

SqlConnection con = new SqlConnection("Data Source=desktop-kr9upqb;Initial Catalog=dataprotector;Integrated Security=True;Pooling=False");

con.Open();

string query = "update loginlog set timeoutoflogin='" + DateTime.Now.ToShortTimeString() + "' where userid='" + ClsUserInfo.userid + "' and dateoflogin='" + ClsUserInfo.dateoflogin + "' and timeinoflogin='"+ClsUserInfo.timeinoflogin + "'";

SqlCommand cmd = new SqlCommand(query, con);

cmd.ExecuteNonQuery();

}

}

}

**ClsUserInfo Program**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace SecurityProject

{

class ClsUserInfo

{

public static string userid;

public static string dateoflogin;

public static string timeinoflogin;

}

}

**Program.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Windows.Forms;

namespace SecurityProject

{

static class Program

{

/// <summary>

/// The main entry point for the application.

/// </summary>

[STAThread]

static void Main()

{

Application.EnableVisualStyles();

Application.SetCompatibleTextRenderingDefault(false);

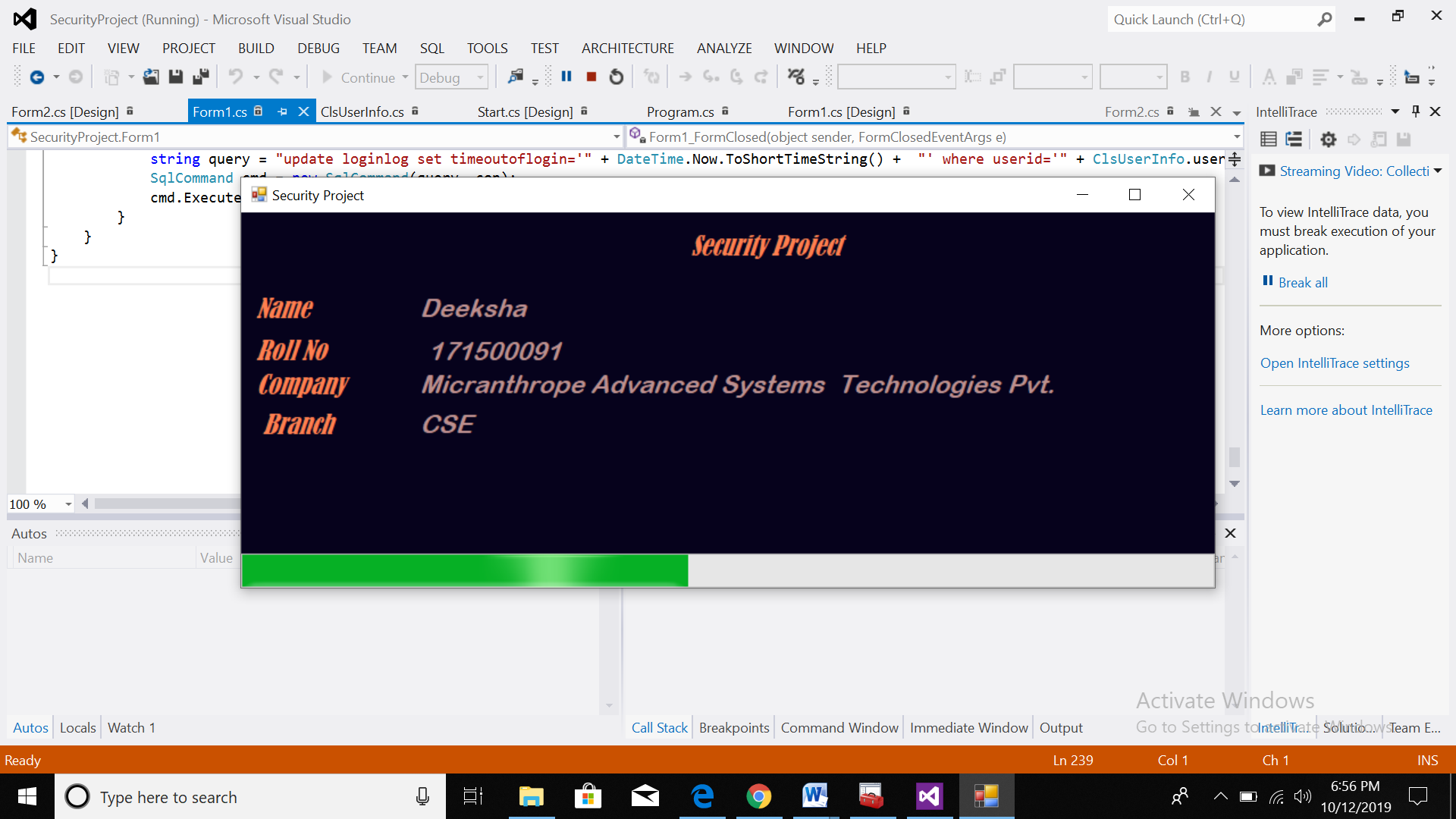
Application.Run(new Form2());

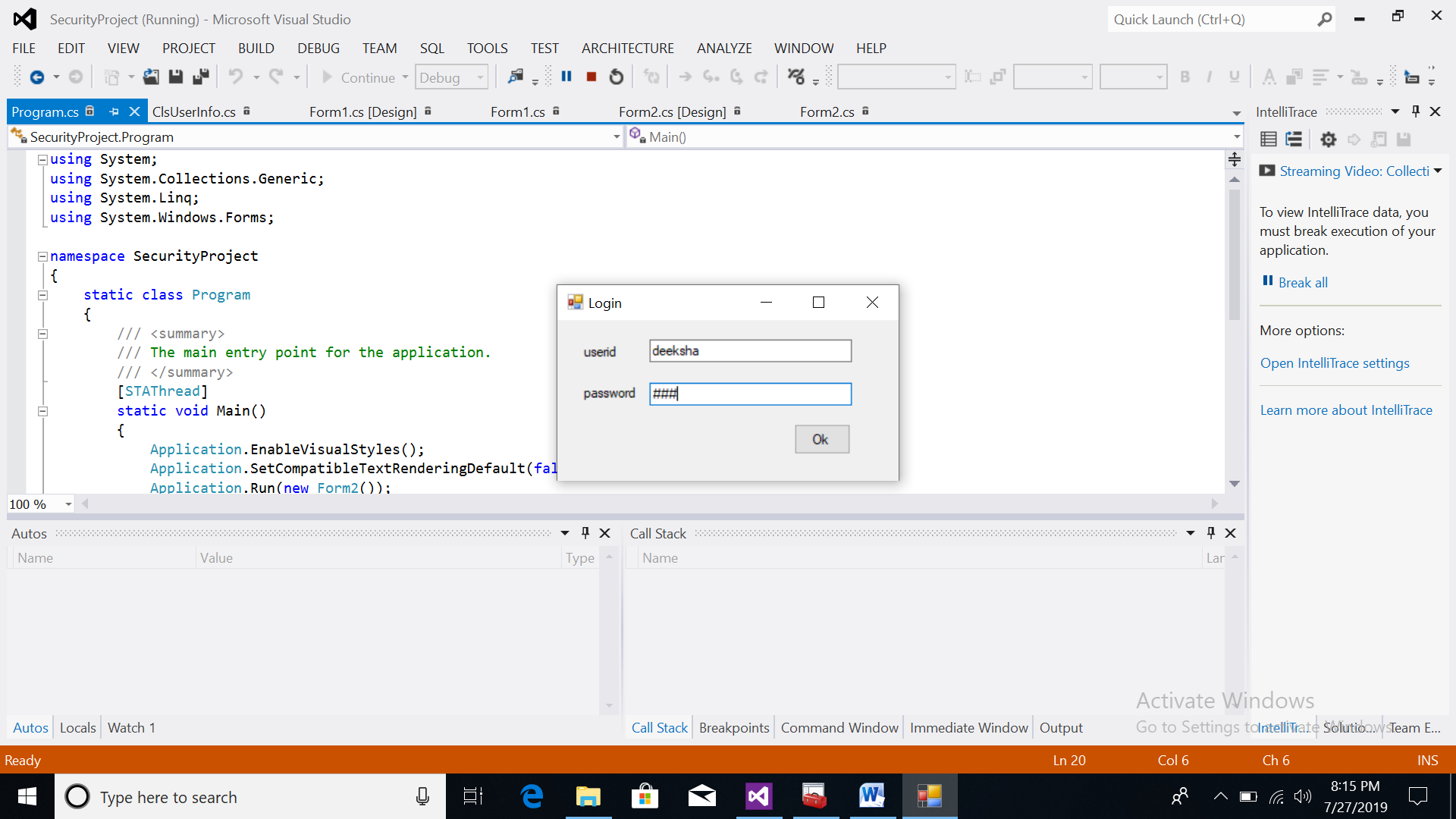
}

}

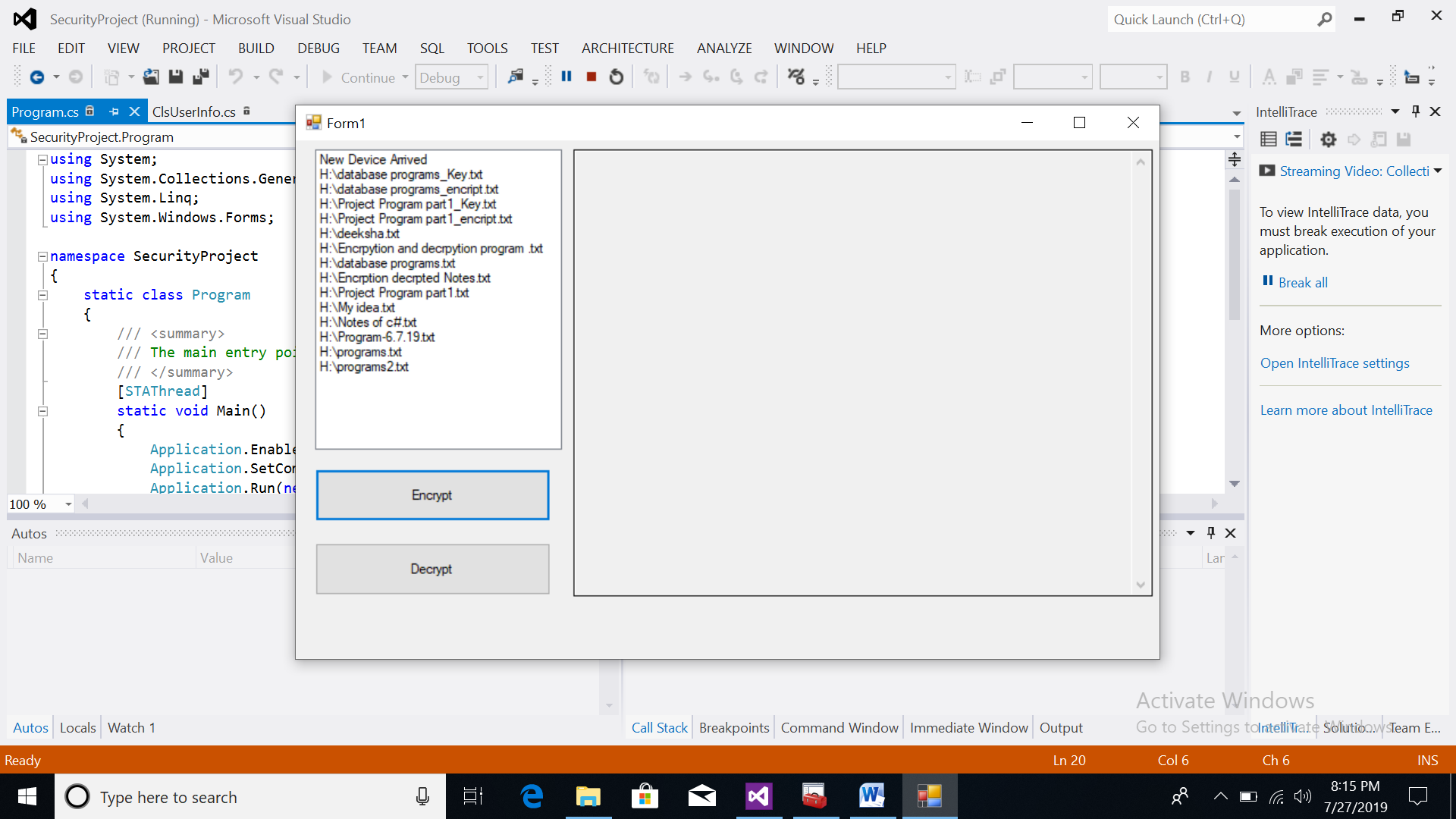
}

**RUN THE PROJECT**

****



**PEN DRIVE INSERT**



**DECRYPED FILE**

