Bahria University,

Karachi Campus



COURSE: Computer Architecture and Logic Design

TERM: FALL 2021, CLASS: BSE- 3 (B)

PROJECT NAME

Social Accounts Password Manager

Made By:

Muhammad Junaid Saleem Qadi

Muhammad Amjad

Muhammad Ameer Hamza

Engr.Dr.Samar Yazdani/ Engr. Ramsha Mashood

Signed Remarks: Score:

**TABLE OF CONTENT: Pg No**

1. INTRODUCTION & PROBLEM
2. TECHNOLOGY
3. FEATURES
4. MODULE DISTRIBUTION
5. CODE
6. INTERFACES
7. CONCLUSION

**Introduction and Problem:**

If you use the same password for more than one web site and any one of them is hacked, you run the risk that someone will be able to use your password to break into your accounts on other sites. For that reason, you should always use a different password for every account and web site you use.

Our password manager that helps you secure and organize your passwords by using the most advanced and secure encryption algorithm (hashing algorithm) to encrypt your passwords information. So no need to remember all those zillions of passwords for each and every website. Our software is an OFFLINE password manager so you can have complete control of your passwords. It also comes with a random password generator to generate unique and strong passwords.

**Technology Used:**

* Java Swing
* C# Windows Form

**Hash Algorithm:**

Hash algorithm is a cryptographic hashing algorithm used to determine the integrity of a particular piece of data. Variations of this algorithm are often used by SSL certificate authorities to sign certificates.

A common application of SHA/MD5 is to encrypting passwords, as the server side only needs to keep track of a specific user’s hash value, rather than the actual password. This is helpful in case an attacker hacks the database, as they will only find the hashed functions and not the actual passwords, so if they were to input the hashed value as a password, the hash function will convert it into another string and subsequently deny access. Additionally, SHAs exhibit the avalanche effect, where the modification of very few letters being encrypted causes a big change in output; or conversely, drastically different strings produce similar hash values. This effect causes hash values to not give any information regarding the input string, such as its original length. In addition, SHAs are also used to detect the tampering of data by attackers, where if a text file is slightly changed and barely noticeable, the modified file’s hash value will be different than the original file’s hash value, and the tampering will be rather noticeable.

**Types of Hashing:**

There are many different types of hash algorithms such as RipeMD, Tiger, xxhash and more, but the most common type of hashing are MD5, SHA-1 and SHA-2.

**1. MD5:**

An MD5 hash is created by taking a string of an any length and encoding it into a 128-bit fingerprint. Encoding the same string using the MD5 algorithm will always result in the same 128-bit hash output. MD5 hashes are commonly used with smaller strings when storing passwords, credit card numbers or other sensitive data in databases such as the popular MySQL. This tool provides a quick and easy way to encode an MD5 hash from a simple string of up to 256 characters in length.

MD5 hashes are also used to ensure the data integrity of files. Because the MD5 hash algorithm always produces the same output for the same given input, users can compare a hash of the source file with a newly created hash of the destination file to check that it is intact and unmodified.

An MD5 hash is NOT encryption. It is simply a fingerprint of the given input. However, it is a one-way transaction and as such it is almost impossible to reverse engineer an MD5 hash to retrieve the original string.

**2. SHA-1:**

Secure Hash Algorithm 1, or SHA-1, was developed in 1993 by the U.S. government's standards agency National Institute of Standards and Technology (NIST). It is widely used in security applications and protocols, including TLS, SSL, PGP, SSH, IPsec, and S/MIME. SHA-1 works by feeding a message as a bit string, and producing a 160-bit hash value known as a message digest.

SHA-1 is still widely used, cryptanalysts in 2005 were able to find vulnerabilities on this algorithm that detrimentally compromised its security. These vulnerabilities came in the form of an algorithm that speedily finds collisions with different inputs, meaning that two distinct inputs map to the same digest. It is more secure than MD5.

As of 2010, many organizations have recommended its replacement by SHA-2 or SHA-3. Companies like Microsoft, Google, or Mozilla have announced that their browsers will stop accepting SHA-1 encryption certificates by 2017.

**3. SHA-2:**

Due to the exposed vulnerabilities of SHA-1, cryptographers modified the algorithm to produce SHA-2, which consists of not one but two hash functions known as SHA-256 and SHA-512, using 32- and 64-bit words, respectively but we implement only SHA-256.

SHA-1 and SHA-2 differ in several ways; mainly, SHA-2 produces 224- or 256-sized digests, whereas SHA-1 produces a 160-bit digest; SHA-2 can also have block sizes that contain 1024 bits, or 512 bits, like SHA-1.

A Brute force attacks on SHA-2 are not as effective as they are against SHA-1 which makes SHA-2 a lot safer against these kinds of attacks.

**Features:**

1. Generate Random Password including upper case, lower case , numbers, and special symbols according to user-defined length.
2. Password Encryption using MD5, SHA-1, and SHA-256 Hash.
3. Dashboard which shows all saved passwords with different website where Name , Email , username , password and website url columns included. User can easily copy password in the clipboard and sorting each column’s Record.
4. User can create new record , delete record, and save record.
5. File System is use at Back-end.
6. Modern GUI design with Images , Sound effects, round button ,etc.
7. User credentials to log in and verify their identities to their accounts . New user SignUp their account.
8. Password Recovery when user forget password.

**Module Distribution:**

**Muhammad Amjad:**

Creates GUI of Forget Password Feature and implement it by using decision tree on JAVA.

**Junaid Saleem Qadri:**

**Muhammad Ameer Hamza:**

**Source Code:**

**SignUp Feature:**

public partial class Register : Form

{

string error;

Filing database = new Filing();

Other\_Detail info = new Other\_Detail();

public Register()

{

InitializeComponent();

}

#region Minimize & Close button

private void close\_Button\_Click(object sender, EventArgs e)

{

this.Close();

}

private void minimize\_Button\_Click(object sender, EventArgs e)

{

this.WindowState = FormWindowState.Minimized;

}

private void close\_Button\_MouseEnter(object sender, EventArgs e)

{

close\_Button.BackColor = Color.DarkRed;

}

private void close\_Button\_MouseLeave(object sender, EventArgs e)

{

close\_Button.BackColor = Color.Transparent;

}

private void minimize\_Button\_MouseEnter(object sender, EventArgs e)

{

minimize\_Button.BackColor = Color.DodgerBlue;

}

private void minimize\_Button\_MouseLeave(object sender, EventArgs e)

{

minimize\_Button.BackColor = Color.Transparent;

}

#endregion

#region Moving Panel through mouse

[DllImport("user32.Dll", EntryPoint = "ReleaseCapture")]

private extern static void ReleaseCapture();

[DllImport("user32.Dll", EntryPoint = "SendMessage")]

private extern static void SendMessage(System.IntPtr hWnd, int wMsg, int wParam, int lparam);

private void panel1\_MouseDown(object sender, MouseEventArgs e)

{

ReleaseCapture();

SendMessage(this.Handle, 0x112, 0xf012, 0);

}

#endregion

private void SignUp\_button\_Click(object sender, EventArgs e)

{

error\_display.ForeColor = Color.Red;

error\_display.Text = error;

if (error == "")

{

bool created = database.create\_Account\_file(UserName.Text, Password.Text, Code.Text,Other\_Detail.instance.fName, Other\_Detail.instance.school, Other\_Detail.instance.city, Other\_Detail.instance.sport, Other\_Detail.instance.loc);

if (created)

{

SignUp\_button.FlatAppearance.MouseDownBackColor = Color.Green;

error\_display.Text = "Your Account Create Succesfully !";

error\_display.ForeColor = Color.DarkSeaGreen;

}

}

}

// for change color of button

private void SignUp\_button\_MouseEnter(object sender, EventArgs e)

{

if (check())

SignUp\_button.FlatAppearance.MouseDownBackColor = Color.Green;

else

SignUp\_button.FlatAppearance.MouseDownBackColor = Color.DarkRed;

}

private bool check()

{

if (UserName.Text == "")

{

error = "UserName is inValid";

return false;

}

if (Code.Text == "")

{

error = "Code is inValid";

return false;

}

if(Password.Text == "" || Password.Text.Length<4)

{

error = "Password is inCorrect";

return false;

}

if(database.user\_Exist(UserName.Text))

{

error = "Sorry Your Account is already Created";

return false;

}

if (!Other\_Detail.instance.allow)

{

error = "Please Provide us Other Info";

return false;

}

error = "";

return true;

}

//reset every thing

private void reset()

{

if (error == "" || error\_display.Text == "Sorry Your Account is already Created")

{

Code.Clear();

Password.Clear();

UserName.Clear();

error\_display.ForeColor = Color.Red;

error\_display.Text = "";

}

}

private void UserName\_Enter(object sender, EventArgs e)

{

reset();

}

private void detail\_Enter(object sender, EventArgs e)

{

Other\_Detail info = new Other\_Detail();

info.Show();

}

}

**Other Detail :**

public partial class Other\_Detail : Form

{

public string fName,school,city,sport,loc;

public bool allow = false;

public static Other\_Detail instance;

public Other\_Detail()

{

InitializeComponent();

instance = this;

}

private void Save\_btn\_Enter(object sender, EventArgs e)

{

if (father\_Name.Text.Count() > 0 && School\_Name.Text.Count() > 0 && City.Text.Count() > 0 && Sport.Text.Count() > 0 && place.Text.Count() > 0)

{

allow = true;

fName = father\_Name.Text;

school = School\_Name.Text;

city = City.Text;

sport = Sport.Text;

loc = place.Text;

}

else

{

allow = false;

error.Visible = true;

}

}

}

**Login Feature:**

public partial class Login : Form

{

Filing database = new Filing();

public Login()

{

InitializeComponent();

}

#region Moving Panel through mouse

[DllImport("user32.Dll", EntryPoint = "ReleaseCapture")]

private extern static void ReleaseCapture();

[DllImport("user32.Dll", EntryPoint = "SendMessage")]

private extern static void SendMessage(System.IntPtr hWnd, int wMsg, int wParam, int lparam);

private void Login\_MouseDown(object sender, MouseEventArgs e)

{

ReleaseCapture();

SendMessage(this.Handle, 0x112, 0xf012, 0);

}

private void panel1\_MouseDown(object sender, MouseEventArgs e)

{

ReleaseCapture();

SendMessage(this.Handle, 0x112, 0xf012, 0);

}

#endregion

private void UserName\_textBox\_Enter(object sender, EventArgs e)

{

if (UserName\_textBox.Text == "USERNAME")

{

UserName\_textBox.Text = "";

UserName\_textBox.ForeColor = Color.LightGray;

}

}

private void UserName\_textBox\_Leave(object sender, EventArgs e)

{

if (UserName\_textBox.Text == "")

{

UserName\_textBox.Text = "USERNAME";

UserName\_textBox.ForeColor = Color.DimGray;

}

}

private void Password\_taxBox\_Enter(object sender, EventArgs e)

{

if (Password\_taxBox.Text == "PASSWORD")

{

Password\_taxBox.Text = "";

Password\_taxBox.ForeColor = Color.LightGray;

Password\_taxBox.UseSystemPasswordChar = true;

}

}

private void Password\_taxBox\_Leave(object sender, EventArgs e)

{

if (Password\_taxBox.Text == "")

{

Password\_taxBox.Text = "PASSWORD";

Password\_taxBox.ForeColor = Color.DimGray;

Password\_taxBox.UseSystemPasswordChar = false;

}

}

#region Minimize & Exit button

private void exit\_button\_Click(object sender, EventArgs e)

{

Application.Exit();

}

private void minimize\_button\_Click(object sender, EventArgs e)

{

this.WindowState = FormWindowState.Minimized;

}

private void exit\_button\_MouseEnter(object sender, EventArgs e)

{

exit\_button.BackColor = Color.IndianRed;

}

private void exit\_button\_MouseLeave(object sender, EventArgs e)

{

exit\_button.BackColor = Color.Transparent;

}

private void minimize\_button\_MouseEnter(object sender, EventArgs e)

{

minimize\_button.BackColor = Color.DarkViolet;

}

private void minimize\_button\_MouseLeave(object sender, EventArgs e)

{

minimize\_button.BackColor = Color.Transparent;

}

#endregion

private void CreateAccount\_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)

{

Register reg = new Register();

reg.Show();

}

private void Login\_button\_Enter(object sender, EventArgs e)

{

string check = database.Account\_Verify(UserName\_textBox.Text, Password\_taxBox.Text);

if (check == "verify")

{

Login\_button.FlatAppearance.MouseDownBackColor = Color.Olive;

errorDisplay.Text = "";

Filing.User\_Name = UserName\_textBox.Text;

Menu dashboard = new Menu();

dashboard.Show();

this.Close();

}

else

{

Login\_button.FlatAppearance.MouseDownBackColor = Color.Red;

errorDisplay.Text = check;

}

}

private void Forget\_Password\_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)

{

ExecuteCommand(@"C:\Project\ProjectBSE3B.jar");

}

private void ExecuteCommand(string command)

{

int exitCode;

ProcessStartInfo processInfo;

Process process;

processInfo = new ProcessStartInfo("cmd.exe", "/c" + command);

processInfo.CreateNoWindow = true;

processInfo.UseShellExecute = false;

processInfo.RedirectStandardOutput = true;

processInfo.RedirectStandardOutput = true;

process = Process.Start(processInfo);

process.WaitForExit();

string output = process.StandardOutput.ReadToEnd();

string error = process.StandardError.ReadToEnd();

exitCode = process.ExitCode;

process.Close();

}

}

**Main Menu Feature:**

public partial class Menu : Form

{

public Menu()

{

InitializeComponent();

}

#region minimize , maximize , close button

private void close\_icon\_Click(object sender, EventArgs e)

{

Application.Exit();

}

int LX, LY;

private void miximize\_icon\_Click(object sender, EventArgs e)

{

LX =this.Location.X;

LY =this.Location.Y;

this.Size = Screen.PrimaryScreen.WorkingArea.Size;

this.Location = Screen.PrimaryScreen.WorkingArea.Location;

miximize\_icon.Visible = false;

maximizer\_small\_icon.Visible = true;

}

private void maximizer\_small\_icon\_Click(object sender, EventArgs e)

{

this.Size = new Size(1400, 650);

this.Location = new Point(LX,LY);

maximizer\_small\_icon.Visible=false;

miximize\_icon.Visible=true;

}

private void minimize\_icon\_Click(object sender, EventArgs e)

{

this.WindowState = FormWindowState.Minimized;

}

#endregion

#region Moving Panel through mouse

[DllImport("user32.Dll", EntryPoint = "ReleaseCapture")]

private extern static void ReleaseCapture();

[DllImport("user32.Dll", EntryPoint = "SendMessage")]

private extern static void SendMessage(System.IntPtr hWnd, int wMsg, int wParam, int lparam);

private void title\_MouseDown(object sender, MouseEventArgs e)

{

ReleaseCapture();

SendMessage(this.Handle, 0x112, 0xf012, 0);

}

#endregion

private void dataPanel(object displayForm)

{

if(this.panel\_display.Controls.Count > 0)

this.panel\_display.Controls.RemoveAt(0);

Form dF = displayForm as Form;

dF.TopLevel = false;

dF.Dock = DockStyle.Fill;

this.panel\_display.Controls.Add(dF);

this.panel\_display.Tag = dF;

dF.Show();

}

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

{

dataPanel(new Display());

}

private void newPassword\_btn\_Click(object sender, EventArgs e)

{

NewPassword pass = new NewPassword();

pass.Show();

}

private void Encrypt\_btn\_Click(object sender, EventArgs e)

{

Encrypt passEncrypt = new Encrypt();

passEncrypt.Show();

}

}

**Dashboard Feature:**

public partial class Display : Form

{

public Display()

{

InitializeComponent();

}

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

{

string[] data\_row;

data\_row = File.ReadAllLines($"{Filing.rootPath}\\{Filing.User\_Name}\\Data.txt");

for(int i = 0; i < data\_row.Length; i++)

{

string[] data = data\_row[i].Split(',');

int line = dataGridView1.Rows.Add();

dataGridView1.Rows[line].Cells[0].Value = data[0];

dataGridView1.Rows[line].Cells[1].Value = data[1];

dataGridView1.Rows[line].Cells[2].Value = data[2];

dataGridView1.Rows[line].Cells[3].Value = data[3];

dataGridView1.Rows[line].Cells[4].Value = data[4];

}

}

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

if(dataGridView1.Columns[e.ColumnIndex].Name == "del\_col")

{

if (MessageBox.Show("Are you sure want to delete this record ?", "Message",

MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.Yes)

{

string name = dataGridView1.Rows[e.RowIndex].Cells["name\_col"].Value.ToString();

string email = dataGridView1.Rows[e.RowIndex].Cells["email\_col"].Value.ToString();

string usr = dataGridView1.Rows[e.RowIndex].Cells["user\_col"].Value.ToString();

string passw = dataGridView1.Rows[e.RowIndex].Cells["pass\_col"].Value.ToString();

string web = dataGridView1.Rows[e.RowIndex].Cells["web\_col"].Value.ToString();

Filing.delete($"{name},{email},{usr},{passw},{web}");

dataGridView1.Rows.RemoveAt(e.RowIndex);

}

}

if (dataGridView1.Columns[e.ColumnIndex].Name == "copy\_col")

{

Clipboard.SetText(dataGridView1.Rows[e.RowIndex].Cells[3].Value.ToString());

}

}

}

**Create New Password :**

public partial class NewPassword : Form

{

public NewPassword()

{

InitializeComponent();

}

private void cancel\_btn\_Click(object sender, EventArgs e)

{

this.Close();

}

private void option\_btn\_Enter(object sender, EventArgs e)

{

Check\_List opt = new Check\_List();

opt.Show();

}

private void generate\_btn\_Click(object sender, EventArgs e)

{

Password.Text = generatePassword(Check\_List.instance.passwordLenght);

}

private string generatePassword(int lenght)

{

StringBuilder sb = new StringBuilder();

Random random = new Random();

int i = 0;

while (i < lenght)

{

try

{

sb.Append(Check\_List.instance.passwordList[random.Next(0,

Check\_List.instance.passwordList.Count) ] );

}

catch (Exception e) { }

i++;

}

return sb.ToString();

}

private void save\_btn\_Enter(object sender, EventArgs e)

{

string filepath = $"{Filing.rootPath}\\{Filing.User\_Name}\\Data.txt";

List<string> data = new List<string>();

if (File.Exists(filepath))

data = File.ReadAllLines(filepath).ToList();

data.Add($"{name.Text},{email.Text},{user.Text},{Password.Text},{web.Text}");

File.WriteAllLines(filepath, data);

this.close();

}

}

**Password Generate Option:**

public partial class Check\_List : Form

{

public List<char> passwordList = new List<char>();

public int passwordLenght = 8;

public static Check\_List instance;

public Check\_List()

{

InitializeComponent();

instance = this;

}

private void password\_setting()

{

if (LowerCase.Checked)

{

for (char c = 'a'; c <= 'z'; c++)

{

passwordList.Add(c);

}

}

if (UpperCase.Checked)

{

for (char c = 'A'; c <= 'Z'; c++)

{

passwordList.Add(c);

}

}

if (Numbers.Checked)

{

for (char c = '0'; c <= '9'; c++)

{

passwordList.Add(c);

}

}

if (Special.Checked)

{

passwordList.Add('?');

passwordList.Add('#');

passwordList.Add('\_');

passwordList.Add('@');

passwordList.Add('!');

passwordList.Add('~');

passwordList.Add('$');

passwordList.Add('#');

passwordList.Add('%');

passwordList.Add('^');

passwordList.Add('&');

passwordList.Add('\*');

passwordList.Add('(');

passwordList.Add(')');

passwordList.Add('|');

passwordList.Add('\\');

passwordList.Add('/');

passwordList.Add(':');

passwordList.Add(';');

passwordList.Add(']');

passwordList.Add('[');

passwordList.Add('{');

passwordList.Add('}');

passwordList.Add('<');

passwordList.Add('>');

passwordList.Add(',');

passwordList.Add('.');

}

}

private void ok\_btn\_Enter(object sender, EventArgs e)

{

password\_setting();

if(Length.Text.Length <= 2)

{

passwordLenght = int.Parse(Length.Text);

Length.Text = "8";

}

this.Close();

}

private void Check\_List\_FormClosing(object sender, FormClosingEventArgs e)

{

password\_setting();

}

}

**Password Encryption Feature:**

public partial class Encrypt : Form

{

public Encrypt()

{

InitializeComponent();

}

#region about hash Algo

private void about\_md5\_Click(object sender, EventArgs e)

{

MessageBox.Show("MD5 (Message-Digest algorithm 5) is a widely used cryptographic hash

function that results in a 128-bit hash value. The 128-bit (16-byte) MD5 hashes typically are

represented as 32-digit hexadecimal numbers.", "About MD5",MessageBoxButtons.OK,

MessageBoxIcon.Information);

}

private void about\_Sha1\_Click(object sender, EventArgs e)

{

MessageBox.Show("SHA-1 (Secure Hash Algorithm 1) is a cryptographic hash function which

takes an input and produces a 160-bit (20-byte) hash value. This hash value is rendered as a

hexadecimal number which is 40 digits long. It is more secure than MD5","About SHA-1",

MessageBoxButtons.OK,MessageBoxIcon.Information);

}

private void about\_sha256\_Click(object sender, EventArgs e)

{

MessageBox.Show("SHA-256 (Secure Hash Algorithm 2) is a cryptographic hash function

which takes an input and produces a 256-bit hash value. This hash value is rendered as a

hexadecimal number which is 40 digits long. It is more secure than SHA-1", "About SHA-256",

MessageBoxButtons.OK, MessageBoxIcon.Information);

}

#endregion

private void hash\_Button\_Click(object sender, EventArgs e)

{

clipBoard.BackColor = Color.Transparent;

int digit = comboBox1.SelectedItem == "Max" ? 0 : int.Parse((string)comboBox1.SelectedItem);

if(password.Text != "")

{

SoundPlayer player = new SoundPlayer(@"audio\button\_click.wav");

player.Play();

if (radioButton1.Checked)

{

Display\_Hash.Text = Encryption.ComputeMD5Hash(password.Text,digit);

}

else if (radioButton2.Checked)

{

Display\_Hash.Text = Encryption.ComputeSha256Hash(password.Text,digit);

}

else if (radioButton3.Checked)

{

Display\_Hash.Text = Encryption.ComputeSha1Hash(password.Text,digit);

}

}

else

{

MessageBox.Show("Enter any Password","Invalid",

MessageBoxButtons.OK,MessageBoxIcon.Error);

}

}

private void Form1\_Load(object sender, EventArgs e)

{

comboBox1.SelectedItem = "Max";

}

#region close button

private void pictureBox2\_Click(object sender, EventArgs e)

{

this.Close();

}

private void pictureBox2\_MouseEnter(object sender, EventArgs e)

{

pictureBox2.BackColor = Color.Maroon;

}

private void pictureBox2\_MouseLeave(object sender, EventArgs e)

{

pictureBox2.BackColor = Color.DeepSkyBlue;

}

#endregion

#region radio Button

private void radioButton1\_Click(object sender, EventArgs e)

{

Display\_Hash.Clear();

clipBoard.BackColor = Color.Transparent;

radioButton3.ForeColor = Color.White;

radioButton1.ForeColor = Color.Yellow;

radioButton2.ForeColor = Color.White;

SoundPlayer player = new SoundPlayer(@"audio\choose.wav");

player.Play();

}

private void radioButton2\_Click(object sender, EventArgs e)

{

Display\_Hash.Clear();

clipBoard.BackColor = Color.Transparent;

radioButton3.ForeColor = Color.White;

radioButton1.ForeColor = Color.White;

radioButton2.ForeColor = Color.Yellow;

SoundPlayer player = new SoundPlayer(@"audio\choose.wav");

player.Play();

}

private void radioButton3\_Click(object sender, EventArgs e)

{

Display\_Hash.Clear();

clipBoard.BackColor = Color.Transparent;

radioButton3.ForeColor = Color.Yellow;

radioButton1.ForeColor = Color.White;

radioButton2.ForeColor = Color.White;

}

private void radioButton3\_MouseClick(object sender, MouseEventArgs e)

{

SoundPlayer player = new SoundPlayer(@"audio\choose.wav");

player.Play();

}

#endregion

#region sound of digit list

private void comboBox1\_DropDown(object sender, EventArgs e)

{

SoundPlayer player = new SoundPlayer(@"audio\list.wav");

player.Play();

}

private void comboBox1\_SelectionChangeCommitted(object sender, EventArgs e)

{

Display\_Hash.Clear();

clipBoard.BackColor = Color.Transparent;

SoundPlayer player = new SoundPlayer(@"audio\select.wav");

player.Play();

}

#endregion

private void password\_KeyPress(object sender, KeyPressEventArgs e)

{

clipBoard.BackColor = Color.Transparent;

Display\_Hash.Clear();

SoundPlayer player = new SoundPlayer(@"audio\keyboard.wav");

player.Play();

}

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

{

if (Display\_Hash.Text != "")

{

SoundPlayer player = new SoundPlayer(@"audio\clipboard.wav");

player.Play();

clipBoard.BackColor = Color.SlateBlue;

Clipboard.SetText(Display\_Hash.Text);

}

}

}

**Hash algo Class:**

internal class Encryption

{

public static string ComputeSha1Hash(string data,int digit)

{

using (SHA1Managed shao = new SHA1Managed())

{

byte[] hash = shao.ComputeHash(Encoding.UTF8.GetBytes(data));

StringBuilder sb = new StringBuilder(hash.Length \* 2);

foreach (byte i in hash)

{

// can be "x2" if you want lowercase

sb.Append(i.ToString("X2"));

}

if(digit == 0)

return sb.ToString();

else

return sb.ToString().Substring(0, digit);

}

}

public static string ComputeSha256Hash(string rawData,int digit)

{

// Create a SHA256

using (SHA256 sha256Hash = SHA256.Create())

{

// ComputeHash - returns byte array

byte[] bytes = sha256Hash.ComputeHash(Encoding.UTF8.GetBytes(rawData));

// Convert byte array to a string

StringBuilder builder = new StringBuilder();

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

{

builder.Append(bytes[i].ToString("x2"));

}

if (digit == 0)

return builder.ToString();

else

return builder.ToString().Substring(0, digit);

}

}

public static string ComputeMD5Hash(string input,int digit)

{

// Use input string to calculate MD5 hash

using (MD5 md5 = MD5.Create())

{

byte[] inputBytes = Encoding.ASCII.GetBytes(input);

byte[] hashBytes = md5.ComputeHash(inputBytes);

// Convert the byte array to hexadecimal string

StringBuilder sb = new StringBuilder();

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

{

sb.Append(hashBytes[i].ToString("X2"));

}

if (digit == 0)

return sb.ToString();

else

return sb.ToString().Substring(0, digit);

}

}

}

**Database Class:**

internal class Filing

{

public static string rootPath = @"C:\Project";

public static string User\_Name;

public Filing()

{

Directory.CreateDirectory(rootPath);

}

public bool create\_Account\_file(string userName,string password,string code,string fName,

string school,string city,string sport,string place)

{

string userName\_path = $"{rootPath}\\{userName}";

if (user\_Exist(userName))

return false;

Directory.CreateDirectory(userName\_path);

File.WriteAllText($"{userName\_path}\\Account.txt", $"{userName}\n{password}\n

{code}\n{fName}\n{school}\n{city}\n{sport}\n{place}");

return true;

}

public bool user\_Exist(string userName)

{

return Directory.Exists($"{rootPath}\\{userName}");

}

public string Account\_Verify(string userName,string password)

{

string Account\_path = $"{rootPath}\\{userName}\\Account.txt";

if (user\_Exist(userName))

{

string[] Account\_file = File.ReadAllLines(Account\_path);

if (Account\_file[1] == password)//match password

{

return "verify";

}

else

return "Wrong Password";

}

return "UserName not Found";

}

public static void delete(string name)

{

string filepath = $"{rootPath}\\{User\_Name}\\Data.txt";

List<string> newdata = new List<string>();

if (File.Exists(filepath))

newdata = File.ReadAllLines(filepath).ToList();

newdata.Remove(name);

File.WriteAllLines(filepath, newdata);

}

}

**Forget Password Feature:**

**Main Class:**

package projectbse3b;

public class ProjectBSE3B {

public static void main(String[] args) {

new Starting();

}

}

**Starting Class:**

package projectbse3b;

import java.awt.Color;

import java.awt.Font;

import javax.swing.\*;

import java.awt.event.\*;

public class Starting extends JFrame implements ActionListener{

JButton nextButton;

JLabel projectTitle;

Starting()

{

this.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

this.setSize(700,350);

projectTitle=new JLabel("Password Recovery");

projectTitle.setForeground(Color.WHITE);

projectTitle.setFont(new Font("Aerial",Font.BOLD,39));

projectTitle.setBounds(150,-90,500,300);

ImageIcon startingIcon=new ImageIcon("D:\\BUKC\\Semester 2\\Object Oriented Programming Lab\\Project\\Images\\start.jpg");

JLabel title=new JLabel("Created By:");

JLabel name1=new JLabel("Muhammad Amjad");

JLabel name2=new JLabel("Muhammad Junaid Saleem Qadri");

JLabel name3=new JLabel("Muhammad Ameer Hamza");

title.setFont(new Font("Aerial",Font.BOLD,45));

title.setForeground(Color.WHITE);

title.setBounds(208,-42,333,333);

name1.setForeground(Color.WHITE);

name1.setFont(new Font("Aerial",Font.PLAIN,45));

name1.setBounds(140,6,500,350);

name2.setForeground(Color.WHITE);

name2.setFont(new Font("Aerial",Font.PLAIN,30));

name2.setBounds(110,6,500,350);

name3.setForeground(Color.WHITE);

name3.setFont(new Font("Aerial",Font.PLAIN,35));

name3.setBounds(140,6,500,350);

JLabel img1=new JLabel();

nextButton=new JButton("Next");

nextButton.setBounds(535,250,100,35);

nextButton.setFont(new Font("Aerial",Font.BOLD, 15));

nextButton.addActionListener(this);

nextButton.setFocusable(false);

img1.add(title);

img1.add(name1);

img1.add(name2);

img1.add(name3);

img1.add(nextButton);

img1.add(projectTitle);

img1.setIcon(startingIcon);

img1.setSize(700,400);

this.add(img1);

this.setLayout(null);

this.setTitle("Password Recovery");

this.setResizable(false);

this.setLocation(300,180);

this.setVisible(true);

int x=1;

while(x==1)

{

name1.setVisible(true);

name2.setVisible(false);

name3.setVisible(false);

try

{

Thread.sleep(1000);

}

catch(Exception e){}

name1.setVisible(false);

name2.setVisible(true);

name3.setVisible(false);

try

{

Thread.sleep(1000);

}

catch(Exception e){}

name1.setVisible(false);

name2.setVisible(false);

name3.setVisible(true);

try

{

Thread.sleep(1000);

}

catch(Exception e){}

}

}

@Override

public void actionPerformed(ActionEvent e) {

if(e.getSource()==nextButton)

{

this.dispose();

new Login();

}

}

}

**Decision Tree Class:**

package projectbse3b;

import java.awt.Color;

import java.awt.Font;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.io.File;

import java.util.List;

import javax.swing.ImageIcon;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JTextField;

public class DecisionTree extends JFrame implements ActionListener

{

int p=0;

boolean pressButton=false;

Node root;

List<String> dataOfFile;

String uName;

String fileName;

String sqPath;

JLabel fLabel;

JTextField fName;

JLabel schoolLabel;

JTextField schoolName;

JLabel cityLabel;

JTextField cityName;

JLabel fSportLabel;

JTextField fSportName;

JLabel fPlaceLabel;

JTextField fPlaceName;

JButton submitButton;

String path;

DecisionTree(String secretCode,List<String> dataOfFile,String fileName,int point,String uName)

{

this.fileName=fileName;

this.uName=uName;

this.dataOfFile=dataOfFile;

root=new Node(secretCode);

root.yes=new Node("Secret Code Matched");

if(secretCode.equals(dataOfFile.get(2)))

{

JOptionPane.showMessageDialog(null,root.yes.data,"Alert",JOptionPane.INFORMATION\_MESSAGE);

new PasswordRecovery2(fileName,dataOfFile,uName);

}

else

{

root.no=new Node("Not Matched");

JOptionPane.showMessageDialog(null,root.no.data,"Alert",JOptionPane.INFORMATION\_MESSAGE);

this.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

this.setSize(700,500);

this.setLayout(null);

JLabel label=new JLabel("Secret Questions");

label.setForeground(Color.WHITE);

label.setFont(new Font("Aerial",Font.BOLD,39));

label.setBounds(180,-120,500,300);

fLabel=new JLabel("Enter Father Name: ");

fLabel.setForeground(Color.WHITE);

fLabel.setFont(new Font("Aerial",Font.PLAIN,25));

fLabel.setBounds(30,-60,500,300);

fName=new JTextField();

fName.setFont(new Font("Aerial",Font.PLAIN,15));

fName.setBounds(270,77,250,30);

schoolLabel=new JLabel("Enter School Name: ");

schoolLabel.setForeground(Color.WHITE);

schoolLabel.setFont(new Font("Aerial",Font.PLAIN,25));

schoolLabel.setBounds(30,0,500,300);

schoolName=new JTextField();

schoolName.setFont(new Font("Aerial",Font.PLAIN,15));

schoolName.setBounds(270,137,250,30);

cityLabel=new JLabel("Enter City Name: ");

cityLabel.setForeground(Color.WHITE);

cityLabel.setFont(new Font("Aerial",Font.PLAIN,25));

cityLabel.setBounds(30,60,500,300);

cityName=new JTextField();

cityName.setFont(new Font("Aerial",Font.PLAIN,15));

cityName.setBounds(270,197,250,30);

fSportLabel=new JLabel("Enter Favorite Sport: ");

fSportLabel.setForeground(Color.WHITE);

fSportLabel.setFont(new Font("Aerial",Font.PLAIN,25));

fSportLabel.setBounds(30,120,500,300);

fSportName=new JTextField();

fSportName.setFont(new Font("Aerial",Font.PLAIN,15));

fSportName.setBounds(270,257,250,30);

fPlaceLabel=new JLabel("Enter Favorite Place: ");

fPlaceLabel.setForeground(Color.WHITE);

fPlaceLabel.setFont(new Font("Aerial",Font.PLAIN,25));

fPlaceLabel.setBounds(30,180,500,300);

fPlaceName=new JTextField();

fPlaceName.setFont(new Font("Aerial",Font.PLAIN,15));

fPlaceName.setBounds(270,317,250,30);

submitButton=new JButton("Submit");

submitButton.setFont(new Font("Aerial",Font.BOLD,20));

submitButton.setBounds(330,377,120,35);

submitButton.setFocusable(true);

submitButton.addActionListener(this);

JLabel img1=new JLabel();

ImageIcon startingIcon=new ImageIcon("D:\\BUKC\\Semester 2\\Object Oriented Programming Lab\\Project\\Images\\start.jpg");

img1.setIcon(startingIcon);

img1.setSize(700,600);

img1.add(label);

img1.add(fLabel);

img1.add(fName);

img1.add(schoolLabel);

img1.add(schoolName);

img1.add(cityLabel);

img1.add(cityName);

img1.add(fSportLabel);

img1.add(fSportName);

img1.add(fPlaceLabel);

img1.add(fPlaceName);

img1.add(submitButton);

this.add(img1);

this.setResizable(false);

this.setTitle("Recovery");

this.setLocation(300,180);

this.setVisible(true);

}

}

@Override

public void actionPerformed(ActionEvent e) {

pressButton = true;

if(e.getSource()==submitButton)

{

String fileName="Account.txt";

FilingSearch fileSearch=new FilingSearch();

fileSearch.searchDirectory(new File("C:\\Project\\"+uName), fileName);

List<String> dataOfFile=fileSearch.dataRead();

int x=1;

if(x==1)

{

if(!(fName.getText().equals("")))

{

fLabel.setForeground(Color.WHITE);

x=1;

if(fName.getText().equals(dataOfFile.get(3)))

{

p++;

}

else

{

p--;

}

}

if(!(schoolName.getText().equals("")))

{

schoolLabel.setForeground(Color.WHITE);

x=1;

if(schoolName.getText().equals(dataOfFile.get(4)))

{

p++;

}

else

{

p--;

}

}

if(!(cityName.getText().equals("")))

{

cityLabel.setForeground(Color.WHITE);

x=1;

if(cityName.getText().equals(dataOfFile.get(5)))

{

p++;

}

else

{

p--;

}

}

if(!(fSportName.getText().equals("")))

{

fSportLabel.setForeground(Color.WHITE);

x=1;

if(fSportName.getText().equals(dataOfFile.get(6)))

{

p++;

}

else

{

p--;

}

}

if(!(fPlaceName.getText().equals("")))

{

fPlaceLabel.setForeground(Color.WHITE);

x=1;

if(fPlaceName.getText().equals(dataOfFile.get(7)))

{

p++;

}

else

{

p--;

}

}

if(fName.getText().equals(""))

{

fLabel.setForeground(Color.RED);

x--;

}

if(schoolName.getText().equals(""))

{

schoolLabel.setForeground(Color.RED);

x--;

}

if(cityName.getText().equals(""))

{

cityLabel.setForeground(Color.RED);

x--;

}

if(fSportName.getText().equals(""))

{

fSportLabel.setForeground(Color.RED);

x--;

}

if(fPlaceName.getText().equals(""))

{

fPlaceLabel.setForeground(Color.RED);

x--;

}

}

if(pressButton && x==1)

{

setQuestionDecision(p,root.no,uName,fileName,dataOfFile);

this.dispose();

}

}

}

public void setQuestionDecision(int point,Node root,String uName,String fileName,List<String> dataFile)

{

if(point<5)

{

JOptionPane.showMessageDialog(null,"Data not matched please re-login with correct username","Alert",JOptionPane.ERROR\_MESSAGE);

root.no=new Node("Data not Matched");

new Login();

}

else

{

root.yes=new Node("Data Matched");

JOptionPane.showMessageDialog(null,root.yes.data,"Alert",JOptionPane.INFORMATION\_MESSAGE);

new PasswordRecovery2(fileName,dataOfFile,uName);

}

}

}

**Password Recovery Class:**

package projectbse3b;

import java.awt.Color;

import java.awt.Font;

import javax.swing.\*;

import java.awt.event.\*;

import java.util.\*;

import java.io.\*;

public class PasswordRecovery extends JFrame implements ActionListener {

JButton nextButton;

JButton backButton;

JTextField id;

JTextField secretCode;

PasswordRecovery(String uName)

{

this.setSize(700,350);

this.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

ImageIcon startingIcon=new ImageIcon("D:\\BUKC\\Semester 2\\Object Oriented Programming Lab\\Project\\Images\\start.JPG");

JLabel projectTitle=new JLabel("Password Recovery");

projectTitle.setForeground(Color.WHITE);

projectTitle.setFont(new Font("Aerial",Font.BOLD,39));

projectTitle.setBounds(150,-90,500,300);

JLabel ForgetPass=new JLabel("Password Recovery");

ForgetPass.setForeground(Color.WHITE);

ForgetPass.setFont(new Font("Aerial",Font.BOLD,30));

ForgetPass.setBounds(180,-50,500,300);

JLabel ID=new JLabel("Enter Username:");

ID.setForeground(Color.WHITE);

ID.setFont(new Font("Aerial",Font.BOLD,25));

ID.setBounds(60,0,500,300);

id=new JTextField(uName);

id.setBounds(260,138,300,28);

id.setEditable(false);

JLabel Secret=new JLabel(" Enter SecretCode:");

Secret.setForeground(Color.WHITE);

Secret.setFont(new Font("Aerial",Font.BOLD,25));

Secret.setBounds(32,60,500,300);

secretCode=new JTextField();

secretCode.setFont(new Font("Aerial",Font.BOLD,15));

secretCode.setBounds(260,197,300,28);

nextButton=new JButton("Next");

nextButton.setBounds(535,260,100,35);

nextButton.setFont(new Font("Aerial",Font.BOLD, 15));

nextButton.setFocusable(false);

nextButton.addActionListener(this);

backButton=new JButton("Back");

backButton.setBounds(70,260,100,35);

backButton.setFont(new Font("Aerial",Font.BOLD, 15));

backButton.setFocusable(false);

backButton.addActionListener(this);

JLabel img1=new JLabel();

img1.setIcon(startingIcon);

img1.setSize(700,400);

img1.add(nextButton);

img1.add(backButton);

img1.add(id);

img1.add(secretCode);

img1.add(projectTitle);

img1.add(ID);

img1.add(Secret);

img1.add(ForgetPass);

this.add(img1);

this.setResizable(false);

this.setTitle("Password Recovery");

this.setLocation(300,180);

this.setVisible(true);

}

@Override

public void actionPerformed(ActionEvent e) {

if(e.getSource()==nextButton)

{

FilingSearch fileSearch = new FilingSearch();

FilingSearch fs = new FilingSearch();

String fileName = "Account.txt";

String path="C:\\Project\\"+id.getText();

fileSearch.searchDirectory(new File(path), fileName);

List<String> dataOfFile = fileSearch.dataRead();

int point=0;

this.dispose();

DecisionTree tree=new DecisionTree(secretCode.getText(),dataOfFile,fileName,point,id.getText());

// if(secretCode.getText().equals(dataOfFile.get(3)))

// {

// //Password Step 2

// this.dispose();

// new PasswordRecovery2(fileName,dataOfFile);

// }

// else

// {

// ImageIcon icon=new ImageIcon("D:\\ac.png");

// String[] responses={"Try another way","Login"};

// int x=JOptionPane.showOptionDialog(null,"Secret code does not match please re-enter with correct username or try another way","Forget Password",JOptionPane.YES\_NO\_OPTION,JOptionPane.WARNING\_MESSAGE,icon,responses,0);

// if(x==1||x==-1)

// {

// this.dispose();

// new Login();

// }

// else

// {

//

// }

// }

}

if(e.getSource()==backButton)

{

this.dispose();

new Login();

}

}

}

**Password Recovery 2 Class:**

package projectbse3b;

import java.awt.Color;

import java.awt.Font;

import javax.swing.\*;

import java.awt.event.\*;

import java.io.\*;

import java.util.\*;

public class PasswordRecovery2 extends JFrame implements ActionListener{

JPasswordField password1;

JPasswordField password2;

JButton confirmButton;

JLabel Pass;

JLabel ConfirmPass;

String fileName;

List<String> dataFile;

String uName;

PasswordRecovery2(String fileName,List<String> dataFile,String uName)

{

this.uName=uName;

this.dataFile=dataFile;

this.fileName=fileName;

this.setSize(700,350);

this.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

ImageIcon startingIcon=new ImageIcon("D:\\BUKC\\Semester 2\\Object Oriented Programming Lab\\Project\\Images\\start.jpg");

JLabel projectTitle=new JLabel("Password Recovery");

projectTitle.setForeground(Color.WHITE);

projectTitle.setFont(new Font("Aerial",Font.BOLD,39));

projectTitle.setBounds(150,-90,500,300);

Pass=new JLabel("Enter Password:");

Pass.setForeground(Color.WHITE);

Pass.setFont(new Font("Aerial",Font.BOLD,25));

Pass.setBounds(60,0,500,300);

password1=new JPasswordField();

password1.setFont(new Font("Aerial",Font.BOLD,15));

password1.setBounds(260,138,300,28);

ConfirmPass=new JLabel("Confirm Password:");

ConfirmPass.setForeground(Color.WHITE);

ConfirmPass.setFont(new Font("Aerial",Font.BOLD,25));

ConfirmPass.setBounds(30,60,500,300);

password2=new JPasswordField();

password2.setFont(new Font("Aerial",Font.BOLD,15));

password2.setBounds(260,197,300,28);

confirmButton=new JButton("Confirm");

confirmButton.setBounds(535,260,100,35);

confirmButton.setFont(new Font("Aerial",Font.BOLD, 15));

confirmButton.setFocusable(false);

confirmButton.addActionListener(this);

JLabel img1=new JLabel();

img1.setIcon(startingIcon);

img1.setSize(700,400);

img1.add(projectTitle);

img1.add(Pass);

img1.add(ConfirmPass);

img1.add(password1);

img1.add(password2);

img1.add(confirmButton);

this.add(img1);

this.setResizable(false);

this.setTitle("Password Recovery");

this.setLocation(300,180);

this.setVisible(true);

}

@Override

public void actionPerformed(ActionEvent e) {

if(e.getSource()==confirmButton)

{

String Pass1=new String(password1.getPassword());

String Pass2=new String(password2.getPassword());

int x=1;

if(x>=1)

{

if(Pass2.equals(Pass1))

{

ConfirmPass.setForeground(Color.WHITE);

x=1;

}

if(!(Pass1.equals("")))

{

Pass.setForeground(Color.WHITE);

x=1;

}

if(Pass1.equals(""))

{

Pass.setForeground(Color.red);

ConfirmPass.setForeground(Color.red);

x--;

}

if(!(Pass2.equals(Pass1)))

{

ConfirmPass.setForeground(Color.red);

x--;

}

}

if(x==1)//Replace Password

{

FilingSearch fileSearch = new FilingSearch();

// String fileName = UName+".txt";

fileSearch.searchDirectory(new File("C:\\Project\\"+uName), fileName);

List<String> dataOfFile = fileSearch.dataRead();

File newFile = new File(fileSearch.path);

try{

FileWriter addingTextInFile = new FileWriter(newFile);

addingTextInFile.append(dataOfFile.get(0)+"\n");

addingTextInFile.append(Pass1+"\n");

addingTextInFile.append(dataOfFile.get(2)+"\n");

addingTextInFile.append(dataOfFile.get(3)+"\n");

addingTextInFile.append(dataOfFile.get(4)+"\n");

addingTextInFile.append(dataOfFile.get(5)+"\n");

addingTextInFile.append(dataOfFile.get(6)+"\n");

addingTextInFile.append(dataOfFile.get(7)+"\n");

addingTextInFile.close();

}catch(Exception ez){

System.out.println(ez.getMessage());

}

dataFile.set(1,Pass1);

String Paths=fileSearch.path.toString();

//Create File in new File and Change Password

this.dispose();

new Login();

}

}

}

}

**Project Interfaces:**

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

A screenshot of a video game

Description automatically generated with medium confidence

**Conclusion:**