using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Security.Cryptography;

using System.IO;

using System.Text;

namespace genera\_identificador

{

public static class Cryptografia

{

public static String Decrypt(string cipherText)

{

string EncryptionKey = "MAKV2SPBNI99212werdiovw45ifvwsivpRFGsindvpwsidn3243#$!4";

cipherText = cipherText.Replace(" ", "+");

byte[] cipherBytes = Convert.FromBase64String(cipherText);

using (Aes encryptor = Aes.Create())

{

Rfc2898DeriveBytes pdb = new Rfc2898DeriveBytes(EncryptionKey, new byte[] { 0x49, 0x76, 0x61, 0x6e, 0x20, 0x4d, 0x65, 0x64, 0x76, 0x65, 0x64, 0x65, 0x76 });

encryptor.Key = pdb.GetBytes(32);

encryptor.IV = pdb.GetBytes(16);

using (MemoryStream ms = new MemoryStream())

{

using (CryptoStream cs = new CryptoStream(ms, encryptor.CreateDecryptor(), CryptoStreamMode.Write))

{

cs.Write(cipherBytes, 0, cipherBytes.Length);

cs.Close();

}

cipherText = Encoding.Unicode.GetString(ms.ToArray());

}

}

return cipherText;

}

public static String Encrypt(string clearText)

{

string EncryptionKey = "MAKV2SPBNI99212werdiovw45ifvwsivpRFGsindvpwsidn3243#$!4";

byte[] clearBytes = Encoding.Unicode.GetBytes(clearText);

using (Aes encryptor = Aes.Create())

{

Rfc2898DeriveBytes pdb = new Rfc2898DeriveBytes(EncryptionKey, new byte[] { 0x49, 0x76, 0x61, 0x6e, 0x20, 0x4d, 0x65, 0x64, 0x76, 0x65, 0x64, 0x65, 0x76 });

encryptor.Key = pdb.GetBytes(32);

encryptor.IV = pdb.GetBytes(16);

using (MemoryStream ms = new MemoryStream())

{

using (CryptoStream cs = new CryptoStream(ms, encryptor.CreateEncryptor(), CryptoStreamMode.Write))

{

cs.Write(clearBytes, 0, clearBytes.Length);

cs.Close();

}

clearText = Convert.ToBase64String(ms.ToArray());

}

}

return clearText;

}

}

}