

"Año del Bicentenario del Perú: 200 años de Independencia"

FACULTAD DE INGENIERÍA Y ARQUITECTURA

ESCUELA PROFESIONAL INGENIERÍA DE SISTEMAS



“INFORME”

ASIGNATURA: Desarrollo de Plataforma de Software

DOCENTE: Ing. Hugo Espetia Huamanga

INTEGRANTES :

- ARTEAGA LOPEZ ABRAHAM
- CCAPA ESPINOZA YOHELMIR
- CUSIHUALLPA CORRALES SEBASTIAN OZZY
- DELGADO PAUCAR FRENSSEN WOLFRAN
- PAULLO OLIVERA MIRCO FABIAN
- SALVA PALOMINO POOL ANDRE

CUSCO-PERÚ

2022

INTRODUCCIÓN

Las consideraciones sobre seguridad se deben aplicar a todo el desarrollo de la aplicación, desde el diseño hasta la implementación. Comenzando desde la ejecución del Visual Studio de la forma más segura posible.

Para poder desarrollar de forma eficaz las aplicaciones seguras, es preciso tener unos conocimientos básicos sobre los conceptos y las características de seguridad de las plataformas para las que se desarrolla. También es necesario que se conozca algunos comandos de codificación.

La mayoría de los errores de codificación que tienen como resultado la aparición de puntos vulnerables de seguridad se producen porque los programadores no hacen suposiciones válidas cuando trabajan con los datos que proporciona el usuario o porque no acaban de entender la plataforma para la que están desarrollando.

DESARROLLO

Creando el Procedimiento Almacenado

```
drop proc spLoginUsuario
create proc spLoginUsuario
@TipoUsuario int,
@Usuario int,
@Contrasena varchar(10)
as
begin
    if @TipoUsuario = 1
    begin
        if exists (select DNI from TPaciente where DNI = @Usuario and Passw = @Contrasena)
        begin
            if exists (select DNI from TPaciente where DNI = @Usuario)
            set @TipoUsuario = 1
            select CodError = 0, Mensaje = @TipoUsuario
        end
        else select CodError = 1, Mensaje = 'Error: Usuario y/o Contraseña Incorrectos'
        end
    if @TipoUsuario = 2
    begin
        if exists (select DNI from TMedico where DNI = @Usuario and Pass = @Contrasena)
        begin
            if exists (select DNI from TMedico where DNI = @Usuario)
            set @TipoUsuario = 2
            select CodError = 0, Mensaje = @TipoUsuario
        end
        else select CodError = 1, Mensaje = 'Error: Usuario y/o Contraseña Incorrectos'
        end
    end
end
go
```

Complementando el boton Cliente

```
2 references
public partial class Login : System.Web.UI.Page
{
    ServiceReference1.LoginSoapClient servicio = new ServiceReference1.LoginSoapClient();
    0 references
    protected void Login1_Authenticate(object sender, AuthenticateEventArgs e)
    {
        int combo = DropDownList1.SelectedIndex;
        int usuario = Convert.ToInt32(Login1.UserName);
        string password = Login1.Password;
        //string[] valores = servicio.Loginn(combo, usuario, password).ToArray();

        if (combo == 2)
        {
            string[] valores = servicio.Loginn(combo, usuario, password).ToArray();
            if (valores[0] == "0" && valores[1] == "Medico")
            {
                FormsAuthentication.RedirectFromLoginPage(usuario.ToString(), false);
                Response.Redirect("~/Usuario.aspx", true);
            }
            else if (valores[0] == "0" && valores[1] == "Paciente")
                Login1.FailureText = "Usted es paciente, no tiene acceso al Sistema";
            else
            {
                Login1.FailureText = valores[1];
            }
        }
        else if (combo == 1)
        {
            string[] valores = servicio.Loginn(combo, usuario, password).ToArray();
            if (valores[0] == "0" && valores[1] == "Medico")
            {
                FormsAuthentication.RedirectFromLoginPage(usuario.ToString(), false);
                Response.Redirect("~/Usuario.aspx", true);
            }
            else if (valores[0] == "0" && valores[1] == "Paciente")
                Login1.FailureText = "Usted es medico, no tiene acceso al Sistema";
            else
                Login1.FailureText = valores[1];
        }
    }
}
```

Realizando el Login Webservice

```
using CapaEntidades;
using CapaNegocios;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Services;

namespace CapaServicios
{
    /// <summary>
    /// Summary description for Login
    /// </summary>
    [WebService(Namespace = "http://tempuri.org/")]
    [WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
    [System.ComponentModel.ToolboxItem(false)]
    // To allow this Web Service to be called from script, using ASP.NET AJAX, uncomment the following line.
    // [System.Web.Script.Services.ScriptService]
    0 references
    public class Login : System.Web.Services.WebService
    {
        [WebMethod]
        0 references
        public string[] Loginn(int tipo, int _Usuario, string _Contrasena)
        {
            Usuario usuario = new Usuario();
            UsuarioBL usuarioBL = new UsuarioBL();
            usuario.tipo=tipo;
            usuario._Usuario = _Usuario;
            usuario._Contrasena=_Contrasena;
            bool CodError = usuarioBL.Login(usuario);
            string[] valores = new string[2];
            if (CodError == true) valores[0] = "0";
            else valores[0] = "1";
            valores[1] = usuarioBL.Mensaje;
            return valores;
        }
    }
}
```

Capa Negocio

```
using CapaDatos;
using CapaEntidades;
using System;
using System.Collections.Generic;
using System.Data;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace CapaNegocios
{
    2 references
    public class UsuarioBL : Interfaces.IUsuario
    {
        private Datos datos = new DatosSQL();
        private string mensaje;

        1 reference
        public string Mensaje
        {
            get { return mensaje; }
        }

        2 references
        public bool Login(Usuario usuario)
        {
            DataRow fila = datos.TraerDataRow("spLoginUsuario", usuario.tipo, usuario._Usuario, usuario._Contrasena);
            byte CodError = Convert.ToByte(fila["CodError"]);
            mensaje = fila["Mensaje"].ToString();
            if (CodError == 0)
                return true;
            else return false;
        }
    }
}
```

CONCLUSIÓN

- ❖ La seguridad eliminará vulnerabilidades dentro del sistema protegiendo la información mediante medidas de seguridad así como el procesamiento de información

REFERENCIAS BIBLIOGRÁFICAS

KathleenDollard (no date) *Seguridad y registro - visual basic, Seguridad y Registro - Visual Basic* | *Microsoft Learn*. Available at:
<https://learn.microsoft.com/es-es/dotnet/visual-basic/developing-apps/programming/computer-resources/security-and-the-registry> (Accessed: November 14, 2022).

Adegeo (no date) *Introducción a la seguridad - windows forms .NET framework, Introducción a la seguridad - Windows Forms .NET Framework* | *Microsoft Learn*. Available at:
<https://learn.microsoft.com/es-es/dotnet/desktop/winforms/security-in-windows-forms-overview?view=netframeworkdesktop-4.8> (Accessed: November 14, 2022).