```
if (Seguridad.pfncObtenerPermisos("SGAGE_NAC ") ¡= true)
        return RedirectToAction("../Home/");
```

Si el usuario logueado no tiene acceso a este botón en el SAS, la aplicación automáticamente lo redirigirá a la vista Home.
Controlando de este manera el acceso a los formularios a través de permisos asignados en el SAS por usuario.
De esta manera en todos los métodos de todos los controladores tienen un nivel de permisos según los accesos que tiene el usuario autenticado.

3.3.2. Permisos a nivel de vista

Cada módulo (Módulo de Planificación, Módulo de Indicadores y Módulo de Agenda), dispone de un menú opciones nombrado "_Layout.cshtml"
Ubicado en \Views\Shared_Layout.cshtml", en este archivo se procederá a restringir el acceso a los botones del menú según los códigos dados por SAS.

A modo de ejemplo se mostrará una porción de código del "_Layout.cshtml" del módulo de Agenda:

```
@if (Seguridad.pfncPuedeAccederA("SGAGE_NAC") == "1")
{
    <li class="@ViewBag.ClassActiveNuevaActividad">
            <a href='@Url.Action("Create", "Reunion")'>
                <i style="font-size:12px;" class="glyphicon glyphicon-
                tasks"></i> 

                Nueva Actividad

                </a>

        </li>

}
@if (Seguridad.pfncPuedeAccederA("SGAGE_ACT") == "1")
{
<li class="@ViewBag.ClassActiveRegistroActividades">
            <a href='@Url.Action("Index", "Reunion")'><i style="font-
size:12px;" class="glyphicon glyphicon-list-alt"></i> Registro de
Actividades</a>
        </li>

}

@if (Seguridad.pfncPuedeAccederA("SGAGE_CAL") == "1")
{
            <li class="@ViewBag.ClassActiveCalendario">
            <a href='@Url.Action("Calendario", "Reunion")'><i
style="font-size:12px;" class="glyphicon glyphicon-
calendar"></i> Calendario</a>
        </li>

}
```

electrosur s.a.

Sistema Integrado de Planeamiento (SIP)

INTEGRACIÓN CON SISTEMA DE ADMINISTRACIÓN DE SEGURIDAD

ÍNDICE DE CONTENIDO

1. Módulos del SIP 4
2. Registro de los módulos en el SAS 4
2.1. Parámetros Módulo de Planificación 4
2.2. Parámetros Módulo de Indicadores 6
2.3. Parámetros Módulo de Agenda 7
3. Configuración en los Módulos Web 7
3.2. Procedimiento de autenticación 9
3.3. Configuración de autorización por niveles 11
3.3.1. Permisos a nivel de métodos por controlador 11
3.3.2. Permisos a nivel de vista 13

1. Módulos del SIP

Los módulos que se han integrado al Sistema de Administración de Seguridad (SAS) son los siguientes:

- Módulo de Planificación
- Módulo de Indicadores
- Módulo de Agenda

En el entregable "*ET 09-CODIGO FUENTE.zip*" es donde se encuentran los módulos adecuados.

2. Registro de los módulos en el SAS

Los módulos Web deberán ser registrados en el sistema SAS, para obtener los parámetros necesarios para su posterior configuración en los módulos web.

Una vez registrados en el SAS se obtienen los siguientes parámetros por perfiles de usuario:

2.1. Parámetros Módulo de Planificación

Tal como se especificó en el Documento ET 04-REQUERIMIENTOS, se definen 2 perfiles de usuario: Administrador y Usuario, en la tabla 1 se muestra el acceso q tiene cada usuario a la lista de opciones (menú) que dispone el módulo de Planificación.

Módulo de Planificación	Perfiles de usuario		Parámetros SAS
MENU PRINCIPAL	ADMINISTRADOR	USUARIO	
Planificación ELS	X	X	SGPLN_PLN
Misión / Visión	X	X	SGPLN_PLN_PMV
Foda	X	X	SGPLN_PLN_FDA
Matriz	X	X	SGPLN_PLN_MTZ
Factores Críticos	X	X	SGPLN_PLN_FCE
Proceso Presupuestal 2017	X	X	SGPLN_PLN_PPA
Inicio	X		SGPLN_INI
Plan estratégico	X		SGPLN_INI_PLE
Perspectiva	X		SGPLN_INI_PER
Crear perspectiva	X		SGPLN_INI_PER_Crear
Listado de perspectivas	X		SGPLN_INI_PER_Listado
Objetivos	X		SGPLN_INI_OBJ
Crear objetivo estratégico	X		SGPLN_INI_OBJ_Crear
Lista de objetivos estratégicos	X		SGPLN_INI_OBJ_Listado
Estrategias	X		SGPLN_INI_EST
Crear estrategia	X		SGPLN_INI_EST_Crear
Listado de estrategias	X		SGPLN_INI_EST_Listado
Componentes	X		SGPLN_INI_COM
Crear tipo de componente	X		SGPLN_INI_COM_tCrear
Lista de tipos de componentes	X		SGPLN_INI_COM_tListado
Crear componente	X		SGPLN_INI_COM_Crear

Lista de componentes	X		SGPLN_INI_COM_Listado
Tipo Mantenimiento	X		SGPLN_INI_MAN
Crear tipo de mantenimiento	X		SGPLN_INI_MAN_Crear
Lista de tipos de mantenimiento	X		SGPLN_INI_MAN_Listado
Areas	X		SGPLN_INI_ARE
Crear área	X		SGPLN_INI_ARE_Crear
Lista de áreas	X		SGPLN_INI_ARE_Listado
Crear sub área	X		SGPLN_INI_ARE_SubAreaCrear
Lista de sub áreas	X		SGPLN_INI_ARE_SubAreaListado
Sistema eléctrico	X		SGPLN_INI_SIE
Crear sistema eléctrico	X		SGPLN_INI_SIE_Crear
Lista de sistemas eléctricos	X		SGPLN_INI_SIE_Listado
Ejecutor	X		SGPLN_INI_EJE
Crear ejecutor	X		SGPLN_INI_EJE_Crear
Lista de ejecutores	X		SGPLN_INI_EJE_Listado
Unidad de medida	X		SGPLN_INI_UME
Crear unidad de medida	X		SGPLN_INI_UME_Crear
Lista de unidades de medida	X		SGPLN_INI_UME_Listado
Visión estratégica	X	X	SGPLN_VIE
Perspectivas	X	X	SGPLN_VIE_PER
Objetivos	X	X	SGPLN_VIE_OBJ
Iniciativas	X	X	SGPLN_VIE_INI
Plan de acción	X	X	SGPLN_PLA
Actividad	X	X	SGPLN_PLA_ACT
Crear Actividad	X	X	SGPLN_PLA_ACT_Act
Crear Sub actividad	X	X	SGPLN_PLA_ACT_sAct
Crear tarea	X	X	SGPLN_PLA_ACT_Tar
Crear Acción	X	X	SGPLN_PLA_ACT_Acc
Lista de actividades	X	X	SGPLN_PLA_LIA
Crear plan operativo	X	X	SGPLN_PLA_CPO
Lista de Planes Operativos	X	X	SGPLN_PLA_LPO
Reportes	X	X	SGPLN_PLA_RPT
Mapeo de Iniciativas	X	X	SGPLN_PLA_RPT_Mapeo
Plan Operativo	X	X	SGPLN_PLA_RPT_POP
Usuario	X	X	
Cambio de contraseña	X	X	
Salir	X	X	

Tabla 1:Parámetros Módulo de Planificación

2.2. Parámetros Módulo de Indicadores

Tal como se especificó en el Documento ET 04-REQUERIMIENTOS, se definen 2 perfiles de usuario: Administrador y Usuario, en la tabla 2 se muestra el acceso q tiene cada usuario a la lista de opciones (menú) que dispone el módulo de Indicadores.

Módulo de Indicadores	Perfiles de usuario		Parámetros SAS
MENU PRINCIPAL	ADMINISTRADOR	USUARIO	
Inicio	X		SGIND_INI
Indicador	X		SGIND_INI_IND
Agregar tipo de indicador	X		SGIND_INI_IND_aTipo
Lista tipo de indicadores	X		SGIND_INI_IND_lTipo
Agregar indicador	X		SGIND_INI_IND_agregar
Lista de indicadores	X		SGIND_INI_IND_Lista
Nivel de variación	X		SGIND_INI_IND_Nivel
Asociación Indicador Objetivo	X		SGIND_INI_IND_AIO
Variable	X		SGIND_VAR
Agregar variable	X		SGIND_VAR_agregar
Listado de variables	X		SGIND_VAR_Listado
Fórmula	X		SGIND_FOR
Agregar fórmula	X		SGIND_FOR_Agregar
Listado de fómulas	X		SGIND_FOR_Listado
Reporte	X		SGIND_RPT
Estado de los indicadores	X		SGIND_RPT_Estado
Relación de indicadores	X		SGIND_RPT_Relacion
Cuadro de mando	X		SGIND_CDM
Por objetivos estratégicos	X		SGIND_CDM_EST
Por objetivos específicos	X		SGIND_CDM_ESP
General	X		SGIND_CDM_General
Agrupación / Proceso	X		SGIND_APR
Sub agrupación / Sub proceso	X		SGIND_SAP
Ingreso de valores por bloque	X		SGIND_IVB
Cierre de nformación	X		SGIND_CDI
Ingreso de valores	X	X	SGIND_ING
Consulta	X	X	SGIND_CON
Reporte Histórico del Indicador	X	X	SGIND_RHI
Usuario	X	X	
Cambio de contraseña	X	X	
Salir	X	X	

Tabla 2: Parámetros Módulo de Indicadores

2.3. Parámetros Módulo de Agenda

Tal como se especificó en el Documento ET 04-REQUERIMIENTOS, se definen 2 perfiles de usuario: Administrador y Usuario, en la tabla 2 se muestra el acceso q tiene cada usuario a la lista de opciones (menú) que dispone el módulo de Agenda.

Módulo Agenda	Perfiles de usuario		Parámetros SAS
MENU PRINCIPAL	ADMINISTRADOR	USUARIO	
Inicio	X		SGAGE_INI
Lugar	X		SGAGE_INI_Lugar
Tipo Reunión	X		SGAGE_INI_TipoReunion
Tipo Importancia	X		SGAGE_INI_TipoImportancia
Nueva actividad	X	X	SGAGE_NAC
Registro de actividades	X	X	SGAGE_ACT
Calendario	X	X	SGAGE_CAL
Usuario	X	X	SGAGE_USU
Actualización de email	X	X	
Cambio de contraseña	X	X	
Salir	X	X	

Tabla 3: Parámetros Módulo de Agenda

3. Configuración en los Módulos Web

Las configuraciones a realizarse en los módulos son de autenticación y autorización.

3.1. Referencias y entorno inicial

Paso 1: Lo primero que se tiene que hacer es agregar las referencias segnetadapter.dll y serviciosutilitario.dll, tal como se visualiza en la imagen 1, cabe mencionar que estas referencias son agregadas por módulo (Módulo de Planificación, Módulo de Indicadores y Módulo de Agenda).

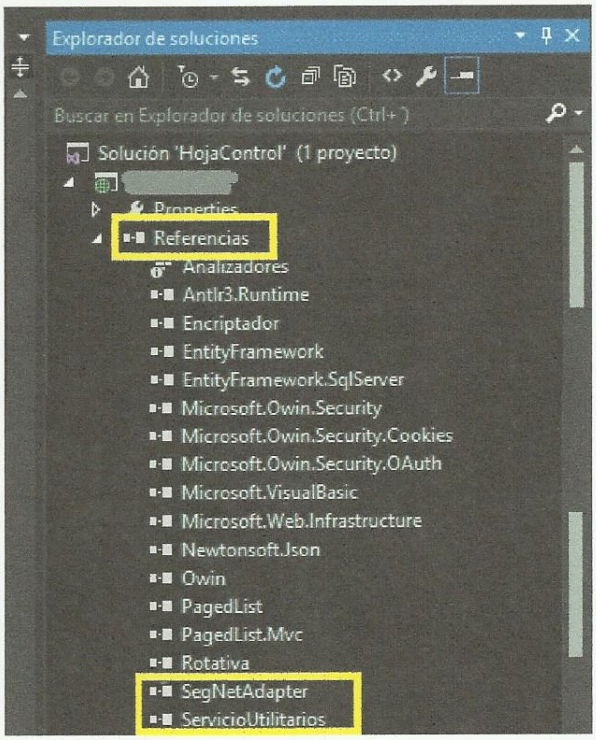
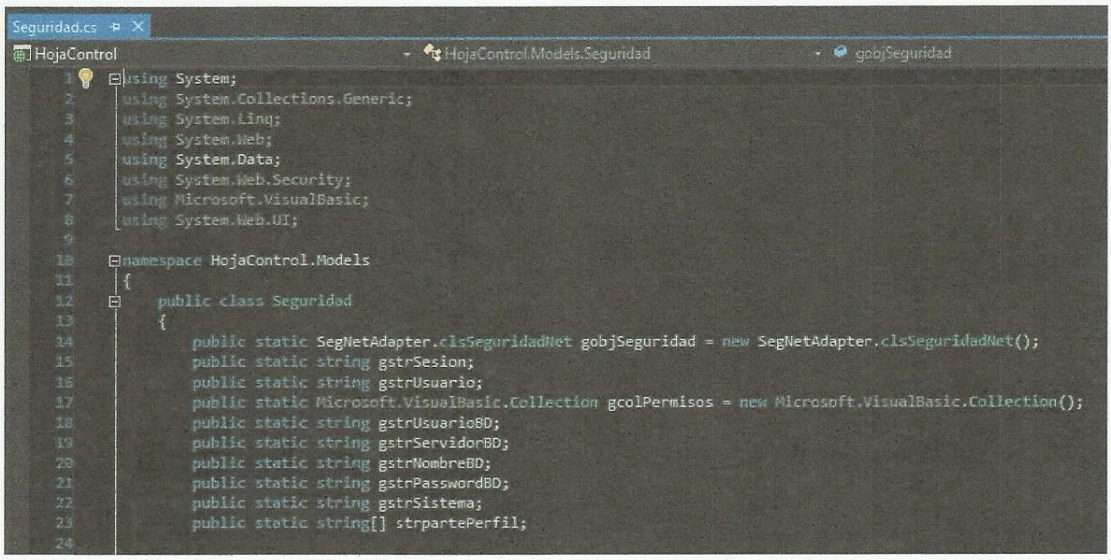


Ilustración 1: Agregar referencias por módulo

Paso 2: A cada módulo (Módulo de Planificación, Módulo de Indicadores y Módulo de Agenda) se agregará la clase seguridad, el cual permitirán establecer variables que permitan el inicio de sesión y permisos a los módulos del sistema.



```
Seguridad.cs
HojaControl                    HojaControl.Models.Seguridad                    gobjSeguridad
1   using System;
2   using System.Collections.Generic;
3   using System.Linq;
4   using System.Web;
5   using System.Data;
6   using System.Web.Security;
7   using Microsoft.VisualBasic;
8   using System.Web.UI;
9
10  namespace HojaControl.Models
11  {
12      public class Seguridad
13      {
14          public static SegNetAdapter.clsSeguridadNet gobjSeguridad = new SegNetAdapter.clsSeguridadNet();
15          public static string gstrSesion;
16          public static string gstrUsuario;
17          public static Microsoft.VisualBasic.Collection gcolPermisos = new Microsoft.VisualBasic.Collection();
18          public static string gstrUsuarioBD;
19          public static string gstrServidorBD;
20          public static string gstrNombreBD;
21          public static string gstrPasswordBD;
22          public static string gstrSistema;
23          public static string[] strpartePerfil;
24
```

Ilustración 2: Extracto de código de la clase Seguridad.cs

Nota 1: este archivo está ubicado en el entregable "*ET 09-CODIGO FUENTE.zip*", por módulo:

- Módulo de Planificación: \SistemaPlanificacion\Models\Seguridad.cs
- Módulo de Indicadores: \cmi\Models\Seguridad.cs

- Módulo de Agenda: \agenda\Models\Seguridad.cs

Nota 2: En la función "fncObtenerValoresPermisosTotal()" se debe hacer los siguientes cambios dependiendo del sistema que se está integrando por primera vez:

```
gstrUsuarioBD = gobjSeguridad.fncObtenerValoresPermisos(gstrUsuario, gstrSistema, gstrSesion, "SGIND_USUARIOBD", "texto");
gstrServidorBD = gobjSeguridad.fncObtenerValoresPermisos(gstrUsuario, gstrSistema, gstrSesion, "SGIND_SERVIDOR", "texto");
gstrNombreBD = gobjSeguridad.fncObtenerValoresPermisos(gstrUsuario, gstrSistema, gstrSesion, "SGIND_BASEDATOS", "texto");
gstrPasswordBD = gobjSeguridad.fncObtenerValoresPermisos(gstrUsuario, gstrSistema, gstrSesion, "SGIND_PASSWORD", "password");
```

Ilustración 3: Extracto de código de la función fncObtenerValoresPermisosTotal()

Las siguientes variables se deben cambiar dependiendo del código del módulo elegido. En la Ilustración 3 se visualiza la configuración del Módulo de Indicadores que tiene el código SGIND, quedando así:

- "SGIND_USUARIOBD"
- "SGIND_SERVIDOR"
- "SGIND_BASEDATOS"
- "SGIND_PASSWORD"

Para el módulo de Planificación:

- "SGPLN _USUARIOBD"
- "SGPLN _SERVIDOR"
- "SGPLN _BASEDATOS"
- "SGPLN _PASSWORD"

Para el módulo de Agenda:

- "SGAGE _USUARIOBD"
- "SGAGE _SERVIDOR"
- "SGAGE _BASEDATOS"
- "SGAGE _PASSWORD"

Nota 3: Estas claves son definidas en el SAS.

Nota 4: Esta configuración ha de realizarse por módulo en cada clase de Seguridad.cs variando en el código del Sistema definido en el SAS.

3.2. Procedimiento de autenticación

El proceso de autenticación, permite configurar un módulo de logueo a través de un formulario de acceso.

Paso 1: Crear un modelo "*LogOnViewModel*.cs"

```
LogOnViewModel.cs
HojaControl                                   HojaCon
1   using System;
2   using System.Collections.Generic;
3   using System.Linq;
4   using System.Web;
5   using System.ComponentModel.DataAnnotations;
6
7   namespace HojaControl.Models
8   {
9       public class LogOnViewModel
10      {
11          [Required]
12          [Display(Name = "Usuario")]
13          public string UserName { get; set; }
14
15          [Required]
16          [DataType(DataType.Password)]
17          [Display(Name = "Contraseña")]
18          public string Password { get; set; }
19
20          [Display(Name = "Confirme su contraseña")]
21          public string NewPassword { get; set; }
22
23          [Display(Name = "Recordarme?")]
24          public bool RememberMe { get; set; }
25      }
26  }
```

Ilustración 4: Modelo LogOnViewModel

Paso 2: Crear la clase "*AccountController*.cs", la cual permitirá recibir la información del formulario y procesarla para el ingreso al módulo respectivo (Módulo de Planificación, Módulo de Indicadores y Módulo de Agenda).

```
AccountController.cs
HojaControl                                   HojaControl.Controllers.AccountController
1   using HojaControl.Models;
2   using System;
3   using System.Collections.Generic;
4   using System.Linq;
5   using System.Web;
6   using System.Web.Mvc;
7   using System.Web.Security;
8   using SegNetAdapter;
9   using ServicioUtilitarios;
10  using Encriptador;
11
12  using Microsoft.AspNet.Identity;
13  using System.Text.RegularExpressions;
14  using System.Threading.Tasks;
15
16  namespace HojaControl.Controllers
17  {
18      public class AccountController : Controller
19      {
20          ProceduresViewsDataContext per = new ProceduresViewsDataContext();
21
22          // GET: /Account/LogOn
23          [AllowAnonymous]
24          public ActionResult LogOn()
25          {
26              LogOnViewModel model = new Models.LogOnViewModel();
```

Ilustración 5: Porción de código, controlador AccountController.cs

Paso 3: Se agregará el formulario web de login (vista) "*LogOn.cshtml*".

```
LogOn.cshtml
1   @model HojaControl.Models.LogOnViewModel
2   @{
3       Layout = null;
4       ViewBag.Title = "Log On";
5
6       ViewBag.ReturnUrl = Request["ReturnUrl"];
7   }
8   <!DOCTYPE html>
9   <html>
10  <head>
11  <META content="IE=edge" http-equiv="X-UA-Compatible">
12      <link href="~/Content/login-box.css" rel="stylesheet" type="text/css" />
13      <link href="~/Content/Site.css" rel="stylesheet" type="text/css" />
14  </head>
15
16  <body>
17
18      @using (Html.BeginForm(null, null, new { returnUrl = ViewBag.ReturnUrl }, FormMethod.Post,
19      {
20          <div style="width:100%; margin-top:5%;">
21              <div id="login-box">
22                  <H2 style="font-size:36px;">Iniciar sesión</H2>
23                  <br />
24                  <span style="font-size: 14px;">Aplicativo Hoja de Control</span>
25                  <br />
26                  <br />
27                  <div id="login-box-name" style="margin-top:20px;">
```

Ilustración 6: Porción de código, Vista LogOn.cshtml

Nota: Los pasos 1, 2 y 3; se tienen que realizar por módulo quedando de la siguiente manera:

Módulo	Modelo	Controlador	Vista
Planificación	LogOnViewModel.cs	AccountController.cs	LogOn.cshtml
Indicadores	LogOnViewModel.cs	AccountController.cs	LogOn.cshtml
Agenda	LogOnViewModel.cs	AccountController.cs	LogOn.cshtml

3.3. Configuración de autorización por niveles

La autorización permite controlar los accesos a cada una de las opciones de cada módulo, esto dándose a nivel de formulario y a nivel de botones (menú).

3.3.1. Permisos a nivel de métodos por controlador

La configuración de los permisos a nivel de formulario se realiza en el menú del módulo de Planificación, menú del módulo de indicadores y menú del módulo de agenda.

Menú del módulo de Planificación

Según la Tabla 1: Parámetros Módulo de Planificación, la primera columna (Menú) visualiza en si la estructura del menú del módulo de Planificación y la cuarta columna visualiza los parámetros obtenidos en el SAS.

A modo de ejemplo se procederá a dar permisos a la opción Perspectiva (Fila 11 de la Tabla 1), para ello se realizará lo siguiente:

- El controlador que contiene a este método es PerspectivaController, en este archivo ubicamos el método Index,

método que permite lista las Perspectivas, dentro de este método se agregará lo siguiente:

```
if (Seguridad.pfncObtenerPermisos("SGPLN_INI_PER ") ¡= true)
        return RedirectToAction("../Home/");
```

Si el usuario logueado no tiene acceso a este botón en el SAS, la aplicación automáticamente lo redirigirá a la vista Home.
Controlando de este manera el acceso a los formularios a través de permisos asignados en el SAS por usuario.
De esta manera en todos los métodos de todos los controladores tienen un nivel de permisos según los accesos que tiene el usuario autenticado.

Menú del módulo de Indicadores

Según la Tabla 2: Parámetros Módulo de Indicadores, la primera columna (Menú) visualiza en si la estructura del menú del módulo de Indicadores y la cuarta columna visualiza los parámetros obtenidos en el SAS.

A modo de ejemplo se procederá a dar permisos a la opción Lista tipo de indicadores (Fila 6 de la Tabla 2), para ello se realizará lo siguiente:

- El controlador que contiene a este método es tipo_indicadorController, en este archivo ubicamos el método Index, método que permite lista los tipos de indicadores, dentro de este método se agregará lo siguiente:

```
if (Seguridad.pfncObtenerPermisos("SGIND_INI_IND_ITipo ") ¡= true)
        return RedirectToAction("../Home/");
```

Si el usuario logueado no tiene acceso a este botón en el SAS, la aplicación automáticamente lo redirigirá a la vista Home.
Controlando de este manera el acceso a los formularios a través de permisos asignados en el SAS por usuario.
De esta manera en todos los métodos de todos los controladores tienen un nivel de permisos según los accesos que tiene el usuario autenticado.

Menú del módulo de Agenda

Según la Tabla 3: Parámetros Módulo de Agenda, la primera columna (Menú) visualiza en si la estructura del menú del módulo de Agenda y la cuarta columna visualiza los parámetros obtenidos en el SAS.

A modo de ejemplo se procederá a dar permisos a la opción Nueva Actividad (Fila 7 de la Tabla 3), para ello se realizará lo siguiente:

- El controlador que contiene a este método es ReunionController, en este archivo ubicamos el método Create, método que permite crear una actividad, dentro de este método se agregará lo siguiente: