

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL



**PROGRAMA DE TECNOLOGÍA EN COMPUTACIÓN Y
DISEÑO GRÁFICO Y PUBLICITARIO**

**PROYECTO DE GRADUACIÓN
PREVIO A LA OBTENCIÓN DEL TÍTULO DE:
ANALISTA DE SISTEMAS**

TEMA

**SISTEMA DE ADMINISTRACIÓN DE PÁGINAS WEB DE
PROTCOM SANTA ELENA**

MANUAL DE ARQUITECTURA

AUTOR

Jessica del Rocío Palacios Yturralde

DIRECTOR

Lcdo. Alex David Espinoza Cárdenas

AÑO

2004-2005

AGRADECIMIENTO

Agradezco mucho a Jehová Dios, por ayudarme a culminar esta etapa de mi vida, siempre ha estado conmigo tanto en las etapas buenas como malas y me ha dado el consejo necesario para poder afrontar todas las dificultades que se me han presentado a lo largo de mi carrera.

Además agradezco a mi papá, mamá, Xavier, Joanita, Joseline, July, Jaimo y Jotu por apoyarme en todo momento. A mis abuelitos por sus buenos consejos. A mis amigos, siempre estuvieron dándome fuerzas para terminar mi carrera y proyecto de graduación, alentándome a no darme por vencida. A mis compañeros de tesis que gracias a su ayuda incondicional hemos podido concluir este proyecto.



DEDICATORIA

Dedico este Proyecto de Graduación a Jehová Dios, a mi familia, a mis amigos, compañeros de tesis por el apoyo, paciencia y dedicación que me han entregado a lo largo de mi carrera universitaria; además por el sacrificio y esfuerzo que me brindaron y siguen brindándome diariamente.

DECLARACIÓN EXPRESA

“La responsabilidad por los hechos, ideas y doctrinas expuestas en este Proyecto de Graduación; y el Patrimonio intelectual del mismo corresponden al **PROTCOM** (*Programa de Tecnología en Computación y Diseño Gráfico*) de la Escuela Superior Politécnica del Litoral”.

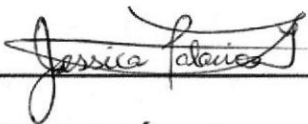
(Reglamentos de exámenes y títulos profesionales de la ESPOL).

FIRMA DEL DIRECTOR DEL PROYECTO DE GRADUACIÓN

A handwritten signature in black ink, consisting of a series of loops and curves, positioned above a horizontal line.

LCDO. ALEX ESPINOZA CÁRDENAS

FIRMA DEL AUTOR DEL PROYECTO DE GRADUACIÓN



JESSICA DEL ROCÍO PALACIOS YTURRALDE

TABLA DE CONTENIDO

1	GENERALIDADES	1
1.1	A quién va dirigido	1
2	AMBIENTE OPERACIONAL DEL SISTEMA ACTUAL	1
2.1	Introducción	1
3	TECNOLOGÍA APLICADA EN EL PORTAL WEB	1
3.1	Descripción Tecnológica	1
3.2	Uso de Paradigmas en la Aplicación Web	2
4	SCRIPTS DEL PORTAL WEB	1
4.1	Creación de la Base de Datos	1
4.1.1	Creación de Tablas	1
4.1.2	Creación de Claves Primarias	22
4.1.3	Creación de Claves Secundarias	28
4.1.4	Creación de Índices	41
4.1.5	Creación de Secuencias	41
4.1.6	Creación de Funciones	46
4.1.7	Creación de Desencadenadores	115
4.1.8	Generación de Datos Constantes	119
4.1.9	Creación de Usuarios predeterminados	146
5	DISTRIBUCIÓN DE LAS CLASES CORRESPONDIENTES A SU NIVEL DE CAPA	1
5.1	Descripción General	1
6	INSTALACIÓN Y CONFIGURACIÓN DE LAS APLICACIONES	1
6.1	Descripción General	1
7	INSTALACIÓN Y CONFIGURACIÓN DEL SERVIDOR PARA LA UTILIZACIÓN DEL SERVICIO LDAP	1
7.1	Introducción a Servicios de Directorios	1
7.2	LightWeight Directory Access Protocol	1
7.2.1	Almacenamiento de la información en el Directorio LDAP	1
7.2.2	Referenciación de la información	3
7.2.3	Acceso a la Información	3
7.2.4	Protección en la información de los accesos no autorizados	3
7.3	Consideraciones Previas	3
7.4	Instalación de los Paquetes SLAPD LDAP-UTILS	3
7.5	Observaciones a la Instalación	4
7.6	Comprobaciones Iniciales de la Instalación	6
7.6.1	Ejecución del demonio	6
7.6.2	Conectando con el Servidor	6

TABLA DE GRÁFICOS

Capítulo 7

Figura 1.	Árbol del directorio LDAP (nombramiento tradicional)	2
Figura 2.	Árbol del directorio LDAP (nombramiento de Internet)	2





CAPÍTULO 1.

Generalidades



En este capítulo, se da una breve explicación de cómo está distribuido este manual por capítulos y para quiénes está dirigido.

1 GENERALIDADES

El **Portal PROTCOM SANTA ELENA** está destinado a promover cualquier tipo de información y de satisfacer las necesidades que poseen todos los usuarios del Internet, pues posee una arquitectura híbrida con una base de datos segmentada que proveerá de los datos suficientes al portal; lo cual lo hará robusto, rápido y flexible.

1.1 A QUIÉN VA DIRIGIDO

Este manual está dirigido al **WebMaster** como un soporte para el desarrollo y mantenimiento del portal, el cual debe tener conocimientos claros del HTML, programación en capas, orientación a objetos, y la tecnología PHP.



CAPÍTULO 2.

Ambiente Operacional del Sistema Actual



En este capítulo, se ofrece una breve explicación de lo que contiene el Portal Web.

2 AMBIENTE OPERACIONAL DEL SISTEMA ACTUAL

2.1 INTRODUCCIÓN

Con este manual, el responsable del mantenimiento del **Portal del PROTCOM Santa Elena** estará en capacidad de conocer:

- Tecnología aplicada para la construcción del Portal Web.
- Scripts del Portal Web.
- Distribución de la programación en cada una de sus clases.

Es importante mencionar que el Portal Web contiene una lógica de presentación de negocios y acceso de datos.

Para la primera capa como proceso cliente, encargada de implementar y demostrar la lógica de presentación, para ésta se usó Internet Explorer versión 5. 5.

Para la segunda capa, es usada para la implementación de un **web server** como el encargado de administrar y ejecutar la codificación embebida en las páginas PHP, y, en el **Transaction Processing Server** como el encargado de administrar e instanciar los componentes encargado de implementar la parte de la regla de negocios y de acceso a la base de datos.

Para implementación de la tercera capa, se usó un **Database Server** como el encargado de permitir la ejecución de los procedimientos almacenados, los cuales contienen la parte restante de la lógica de acceso a los datos.





CAPÍTULO 3.

Tecnología aplicada para la construcción del Portal Web.

En este capítulo, se narra como se encuentra conformado el Portal Web.

3 TECNOLOGÍA APLICADA EN EL PORTAL WEB

3.1 DESCRIPCIÓN TECNOLÓGICA

En esta sección mostraremos cómo fue desarrollado el Portal Web aplicando las diferentes tecnologías que tenemos disponibles en el mundo de la Informática hoy en día y de esta forma cubrir la mayor parte de las necesidades de los usuarios en la navegación a través de la gran red, el **INTERNET**.

Para lograr este objetivo fue necesario hacer un amplio estudio del mercado nacional e internacional, buscando tendencias de navegación y de creación de Sitios y Portales Web. También se realizó un estudio de las Líneas Gráficas sustentando cada color y representación del Diseño Web creado por los Diseñadores Gráficos y así presentar un entorno de impacto en la identificación del usuario con el Portal.

El diseño del Portal Web está basado en el Modelo Lenguaje Unificado o **UML**, aplicando diferentes técnicas donde cada una tiene diferentes ventajas. Para este diseño se han tomados los casos de uso y los diagramas de clase para la construcción de modelos de dominio.

Una vez cubierto el **Modelo de Dominio** (el cual ya fue descrito en forma general) está cubierta la parte de conocer los elementos que van a interactuar con el sistema y portal, entonces podemos avanzar con la creación del **Modelo de Datos** y vaciar toda la información en el mismo, de esta forma creamos una plantilla base que cumpla con todas las normas, estándares y requerimientos para la construcción de cualquier Sitio o Portal Web del Internet haciendo de ellos rápidos, confiables y extensibles. Así el usuario puede acceder al Sistema o Portal Web a través de su espacio personal donde encontrará clasificados por grupos de usuarios todos los servicios que él requiera.

Luego de realizar los estudios iniciales se analizó el tipo de arquitectura a implementar y se escogió la basada en **Cliente/Servidor** haciendo uso del modelo en **n-capas** con Orientación a Objetos (**POO**) brindando así esquemas (**RPC**) basados en motores (**RDBMS**) permitiendo la construcción de cada página web con **PHP** quien nos permite tener un control total de la información que se va a desplegar en el HTML usando el medio "on the fly".

Con el uso de esta tecnología la aplicación brinda la máxima seguridad, disponibilidad, rendimiento de todas las transacciones de envío y recepción de la información; pues para ello se usó una interfaz de tal magnitud que ofrece un acceso de gran flexibilidad de uso y de navegación para los usuarios.

3.2 USO DE PARADIGMAS EN LA APLICACIÓN WEB

El objetivo principal del portal es ofrecer la más amplia y variada información del medio, ubicándose a nivel de la tecnología y de los servicios que ofrece la institución, mostrándola como un ente competitivo y con un gran alcance tecnológico de punta.

Para ello fueron usados los paradigmas que corresponden a una economía de red, los cuales se detallan a continuación:

Primero:	de audiencia a usuario
Segundo:	de medio a contenido
Tercero:	de soporte/formato a multimedia
Cuarto:	de periodicidad a tiempo real
Quinto:	de escasez a abundancia
Sexto:	de intermediación a desintermediación
Séptimo:	de distribución a acceso
Octavo:	de unidireccionalidad a interactividad
Noveno:	de lineal a hipertexto
Décimo:	de información a conocimiento



CAPÍTULO 4.

Scripts del Portal Web.

En este capítulo, se muestran las sentencias SQL básicas para la creación del Portal Web.



4 SCRIPTS DEL PORTAL WEB

4.1 CREACIÓN DE LA BASE DE DATOS

A continuación, se detallan los scripts para la creación de la base de datos correspondiente al portal web que será administrado por un sistema central (SAAP), cuyo desglose de funcionamiento se encuentra detallado en el manual de diseño.

4.1.1 Creación de tablas

```
DROP TABLE SAPW_Buzon CASCADE;
CREATE TABLE SAPW_Buzon (
    BuSu_Codigo      INT                NOT NULL,
    Esta_Codigo      INT                NOT NULL,
    Pers_Cedula      VARCHAR(10)        NOT NULL,
    BuSu_EnviadoPor   VARCHAR(60)        NOT NULL,
    BuSu_Mail         VARCHAR(100),
    BuSu_FechaHora    VARCHAR(20)        NOT NULL,
    BuSu_Sugerencia   VARCHAR(10000)     NOT NULL,
    Esta_Buzon        INT                NOT NULL,
    BuSu_Tipo         INT                NOT NULL,
    BuSu_LeidoCoordinador VARCHAR(1)     NOT NULL
    CHECK (BuSu_LeidoCoordinador = 'S' OR BuSu_LeidoCoordinador = 'N'),
    BuSu_LeidoWebMaster VARCHAR(1)      NOT NULL
    CHECK (BuSu_LeidoWebMaster = 'S' OR BuSu_LeidoWebMaster = 'N')
);
```

```
DROP TABLE SAPW_Curriculum_Vitae CASCADE;
CREATE TABLE SAPW_Curriculum_Vitae (
    Pers_Cedula      VARCHAR(10)        NOT NULL,
    CuVi_EstudiosPrimarios VARCHAR(10000) NOT NULL,
    CuVi_EstudiosSecundarios VARCHAR(10000) NOT NULL,
    CuVi_EstudiosSuperiores VARCHAR(10000) NOT NULL,
    CuVi_OtrosEstudios  VARCHAR(10000) NOT NULL,
    CuVi_ExperienciaLaboral VARCHAR(10000) NOT NULL,
    CuVi_ReferenciaPersonal VARCHAR(10000) NOT NULL
);
```

```
DROP TABLE SAPW_Horario_Paralelo CASCADE;
CREATE TABLE SAPW_Horario_Paralelo(
    Hora_Codigo INT                NOT NULL,
    NiAc_Codigo INT                NOT NULL,
    Carr_Codigo INT                NOT NULL,
    Mate_Codigo VARCHAR(10)        NOT NULL,
    Para_Paralelo INT              NOT NULL,
    Anio_Codigo INT                NOT NULL,
    Peri_Codigo INT                NOT NULL,
    Hopa_Dia      INT              NOT NULL
```



```

CHECK (HoPa_Dia = 1 OR HoPa_Dia = 2 OR
      HoPa_Dia = 3 OR HoPa_Dia = 4 OR
      HoPa_Dia = 5 OR HoPa_Dia = 6 OR
      HoPa_Dia = 7),
HoPa_Aula INT NOT NULL,
Tipo_Codigo INT NOT NULL
);

```

DROP TABLE SAPW_Reservacion_Laboratorio CASCADE;

```

CREATE TABLE SAPW_Reservacion_Laboratorio (
  ReLa_Numero INT NOT NULL,
  Usua_Login VARCHAR(40) NOT NULL,
  Labo_Codigo INT NOT NULL,
  ReLa_Fecha DATE NOT NULL,
  ReLa_FechaReservacion DATE NOT NULL,
  ReLa_HoraReservacionIni TIME NOT NULL,
  ReLa_HoraReservacionFin TIME NOT NULL,
  ReLa_Motivo VARCHAR(10000) NOT NULL,
  ReLa_ProyectorSlides VARCHAR(1) NOT NULL,
  CHECK (ReLa_ProyectorSlides = 'S' OR ReLa_ProyectorSlides = 'N'),
  ReLa_ProyectorAcetatos VARCHAR(1) NOT NULL,
  CHECK (ReLa_ProyectorAcetatos = 'S' OR ReLa_ProyectorAcetatos = 'N'),
  ReLa_DataShowInfocus VARCHAR(1) NOT NULL,
  CHECK (ReLa_DataShowInfocus = 'S' OR ReLa_DataShowInfocus = 'N'),
  ReLa_Computador VARCHAR(1) NOT NULL,
  CHECK (ReLa_Computador = 'S' OR ReLa_Computador = 'N'),
  ReLa_Auditorio VARCHAR(1) NOT NULL,
  CHECK (ReLa_Auditorio = 'S' OR ReLa_Auditorio = 'N'),
  ReLa_Otros VARCHAR(1) NOT NULL,
  CHECK (ReLa_Otros = 'S' OR ReLa_Otros = 'N'),
  ReLa_TV_VHS VARCHAR(1) NOT NULL,
  CHECK (ReLa_TV_VHS = 'S' OR ReLa_TV_VHS = 'N'),
  ReLa_EquipoFotografico VARCHAR(1) NOT NULL,
  CHECK (ReLa_EquipoFotografico = 'S' OR ReLa_EquipoFotografico = 'N'),
  Esta_Codigo INT NOT NULL
);

```

DROP TABLE SAPW_Pasantia CASCADE;

```

CREATE TABLE SAPW_Pasantia(
  SoPa_codigo INT NOT NULL,
  Depa_Codigo INT NOT NULL,
  Esta_Codigo INT NOT NULL,
  Pasa_Descripcion VARCHAR(60) NOT NULL,
  Pasa_FechaInicio DATE NOT NULL,
  Pasa_FechaFin DATE NOT NULL
);

```



```
DROP TABLE SAPW_Solicitud_Pasantia CASCADE;
CREATE TABLE SAPW_Solicitud_Pasantia(
    SoPa_Codigo      INT          NOT NULL,
    Depa_Codigo      INT          NOT NULL,
    Usua_Login       VARCHAR(40)  NOT NULL,
    Esta_Codigo      INT          NOT NULL,
    SoPa_Fecha       DATE         NOT NULL,
    SoPa_Descripcion VARCHAR(10000) NOT NULL
);
```

```
DROP TABLE SAPW_Pasantias_Empresa CASCADE;
CREATE TABLE public.sapw_pasantias_empresa (
    PaSem_Codigo      INT          NOT NULL,
    Empr_RUC          VARCHAR(15)  NOT NULL,
    Depa_Codigo       INT,
    PaSem_Fechainicio DATE,
    PaSem_Fechafin    DATE,
    Carr_Codigo       INT,
);
```

```
DROP TABLE SAPW_Contenido CASCADE;
CREATE TABLE SAPW_Contenido(
    Cont_Codigo      INT          NOT NULL,
    Tipo_Codigo      INT          NOT NULL,
    Usua_Login       VARCHAR(40)  NOT NULL,
    Cont_Descripcion  VARCHAR(60)  NOT NULL,
    Cont_Fecha       DATE         NOT NULL
);
```

```
DROP TABLE SAPW_Respuesta CASCADE;
CREATE TABLE SAPW_Respuesta (
    Resp_Numero      INT          NOT NULL,
    Tema_Numero      INT          NOT NULL,
    Clas_Numero      INT          NOT NULL,
    Foru_Numero      INT          NOT NULL,
    Resp_Persona     VARCHAR(100) NOT NULL,
    Esta_Codigo      INT          NOT NULL,
    Resp_Respuesta   VARCHAR(10000) NOT NULL,
    Resp_Fecha       DATE         NOT NULL
);
```

```
DROP TABLE SAPW_TemaForum CASCADE;
CREATE TABLE SAPW_TemaForum (
    Tema_Numero      INT          NOT NULL,
    Clas_Numero      INT          NOT NULL,
    Foru_Numero      INT          NOT NULL,
    Esta_Codigo      INT          NOT NULL,
    Tema_Pregunta    VARCHAR(100) NOT NULL,
    Tema_Descripcion  VARCHAR(10000) NOT NULL,
    Tema_Fecha       VARCHAR(100) NOT NULL,
);
```

```
Tema_Lecturas      INT      NOT NULL,
Tema_Persona       VARCHAR(100) NOT NULL
);
```

```
DROP TABLE SAPW_Clasificacion CASCADE;
```

```
CREATE TABLE SAPW_Clasificacion (
  Clas_Numero      INT      NOT NULL,
  Foru_Numero      INT      NOT NULL,
  Clas_Nombre      VARCHAR(10000) NOT NULL,
  Clas_Descripcion VARCHAR(10000) NOT NULL);
```

```
DROP TABLE SAPW_Usuario_Forum CASCADE;
```

```
CREATE TABLE SAPW_Usuario_Forum (
  Usua_Login       VARCHAR(40) NOT NULL,
  Foru_Numero      INT      NOT NULL
);
```

```
DROP TABLE SAPW_Forum CASCADE;
```

```
CREATE TABLE SAPW_Forum (
  Foru_Numero      INT      NOT NULL,
  Foru_Nombre      VARCHAR(60) NOT NULL,
  Foru_Fecha       DATE     NOT NULL,
  Foru_Visita      INT      NOT NULL
);
```

```
DROP TABLE SAPW_Empresa CASCADE;
```

```
CREATE TABLE SAPW_Empresa(
  Pers_Cedula      VARCHAR(10) NOT NULL,
  Empr_Ruc         VARCHAR(15) NOT NULL,
  Empr_Nombre      VARCHAR(40) NOT NULL,
  Empr_Direccion   VARCHAR(40) NOT NULL,
  Empr_Telefono    VARCHAR(9)  NOT NULL,
  Empr_Email       VARCHAR(100) NOT NULL
);
```

```
DROP TABLE SAPW_Partес_Pagina CASCADE;
```

```
CREATE TABLE SAPW_Partес_Pagina(
  PaPa_Codigo      INT      NOT NULL,
  CoAn_Codigo      INT      NOT NULL,
  EsAn_Codigo      INT      NOT NULL
);
```

```
DROP TABLE SAPW_Colores_Animo CASCADE;
```

```
CREATE TABLE SAPW_Colores_Animo(
  CoAn_Codigo      INT      NOT NULL,
  EsAn_Codigo      INT      NOT NULL,
  CoAn_CodificacionHex VARCHAR(60) NOT NULL,
  CoAn_Descripcion VARCHAR(60) NOT NULL
);
```

```
DROP TABLE SAPW_Sonido CASCADE;
CREATE TABLE SAPW_Sonido(
    Soni_Codigo      INT          NOT NULL,
    EsAn_Codigo      INT          NOT NULL,
    Soni_Direccion   VARCHAR(10000) NOT NULL
);
```

```
DROP TABLE SAPW_Estado_Animo CASCADE;
CREATE TABLE SAPW_Estado_Animo(
    EsAn_Codigo      INT          NOT NULL,
    EsAn_Descripcion VARCHAR(60)   NOT NULL,
    EsAn_Visita      INT          NOT NULL
);
```

```
DROP TABLE SAPW_Evento CASCADE;
CREATE TABLE SAPW_Evento(
    Even_Codigo      INT          NOT NULL,
    Even_Descripcion VARCHAR(10000) NOT NULL,
    Even_Fecha       DATE         NOT NULL,
    Even_Lugar       VARCHAR(60)   NOT NULL,
    Esta_Codigo      INT          NOT NULL
);
```

```
DROP TABLE SAPW_Codigo_Fuente CASCADE;
CREATE TABLE SAPW_Codigo_Fuente(
    CoFu_Codigo      INT          NOT NULL,
    Esta_Codigo      INT          NOT NULL,
    Usua_Login       VARCHAR(40)   NOT NULL,
    CoFu_Ruta        VARCHAR(100)   NOT NULL,
    CoFu_Titulo      VARCHAR(60)   NOT NULL,
    CoFu_Lenguaje    VARCHAR(60)   NOT NULL,
    CoFu_VotoSiFunciona INT        NOT NULL,
    CoFu_VotoNoFunciona INT        NOT NULL,
    CoFu_Ranking     INT          NOT NULL,
    CoFu_Descripcion VARCHAR(10000) NOT NULL,
    CoFu_Extension   INT,
    CoFu_Pesokb      VARCHAR(10),
);
```

```
DROP TABLE SAPW_Politica_Reservacion CASCADE;
CREATE TABLE SAPW_Politica_Reservacion (
    PoRe_Codigo      INT          NOT NULL,
    PoRe_Descripcion VARCHAR(10000) NOT NULL
);
```

```
DROP TABLE SAPW_Diccionario CASCADE;
CREATE TABLE SAPW_Diccionario (
    Dicc_Codigo      INT          NOT NULL,
    Dicc_Titulo      VARCHAR(40)   NOT NULL,
    Dicc_Frase       VARCHAR(10000) NOT NULL,
);
```

Dicc_Comentario	VARCHAR(10000)	NOT NULL,
Dicc_Url	VARCHAR(1000)	NOT NULL,
Dicc_Incidencia	INT	NOT NULL,
Esta_Codigo	INT	NOT NULL

);

DROP TABLE SAPW_Parametro CASCADE;

CREATE TABLE SAPW_Parametro (

Para_Unidad	VARCHAR(60)	NOT NULL,
Para_Direccion	VARCHAR(60)	NOT NULL,
Para_Telefono	VARCHAR(9)	NOT NULL,
Para_PresentarCostosCert	VARCHAR(1)	NOT NULL
CHECK (Para_PresentarCostosCert = '0' OR Para_PresentarCostosCert = '1'),		
Para_PresentarCostosDipl	VARCHAR(1)	NOT NULL
CHECK (Para_PresentarCostosDipl = '0' OR Para_PresentarCostosDipl = '1'),		
Pers_Cedula	VARCHAR(10)	NOT NULL,
Para_FechaInauguracion	DATE	NOT NULL,
Para_FechaUltimaActualizacion	DATE	NOT NULL,
Para_TiempoAlumnosPadresI	INT	NOT NULL,
Para_TiempoDocentesCampusI	INT	NOT NULL,
Para_TiempoNegociosComunidadI	INT	NOT NULL,
Para_TiempoE_Education	INT	NOT NULL,
Para_TiempoAlumnosPadresE	INT	NOT NULL,
Para_TiempoDocentesCampusE	INT	NOT NULL,
Para_TiempoNegociosCominidade	INT	NOT NULL,
Para_ServidorSMTP	VARCHAR(1000)	NOT NULL,
Para_ServidorLDAP	VARCHAR(1000)	NOT NULL,
Para_ClaveLDAP	VARCHAR(100)	NOT NULL

);

DROP TABLE SAPW_Detalle_Auditoria CASCADE;

CREATE TABLE SAPW_Detalle_Auditoria (

DeAu_Linea	INT	NOT NULL,
CaAu_Codigo	INT	NOT NULL,
DeAu_Descripcion	VARCHAR(10000)	NOT NULL,
DeAu_TablaAcceso	VARCHAR(10000)	NOT NULL,
DeAu_Fecha	TIMESTAMP	NOT NULL

);

DROP TABLE SAPW_Cabecera_Auditoria CASCADE;

CREATE TABLE SAPW_Cabecera_Auditoria (

CaAu_Codigo	INT	NOT NULL,
Usua_Login	VARCHAR(40)	NOT NULL,
CaAu_UbicacionPC	VARCHAR(10000)	NOT NULL,
CaAu_FechaInicioSesion	TIMESTAMP	NOT NULL,
CaAu_FechaFinSesion	TIMESTAMP,	
Esta_codigo	INT	NOT NULL

);

```
DROP TABLE SAPW_Recurso_Humano CASCADE;
CREATE TABLE SAPW_Recurso_Humano (
    ReHu_Codigo          INT          NOT NULL,
    ReHu_Encabezado      VARCHAR(10000) NOT NULL,
    ReHu_Conocimiento    VARCHAR(10000) NOT NULL,
    ReHu_Candidatos      VARCHAR(10000) NOT NULL
);
```

```
DROP TABLE SAPW_Catalogo_Protcom CASCADE;
CREATE TABLE SAPW_Catalogo_Protcom(
    CaPr_Codigo          INT          NOT NULL,
    CaPr_Descripcion     VARCHAR(10000) NOT NULL,
    CaPr_FechaPublicacion DATE        NOT NULL,
    CaPr_URL             VARCHAR(1000) NOT NULL
);
```

```
DROP TABLE SAPW_Anuncio CASCADE;
CREATE TABLE SAPW_Anuncio(
    Anun_Codigo          INT          NOT NULL,
    Esta_Codigo          INT          NOT NULL,
    Anun_Titulo          VARCHAR(60)   NOT NULL,
    Anun_Detalle         VARCHAR(10000) NOT NULL,
    Anun_FechaPublicacion DATE        NOT NULL,
    Anun_Foto            INT
);
```

```
DROP TABLE SAPW_Solicitud_Personal CASCADE;
CREATE TABLE SAPW_Solicitud_Personal(
    SoPe_Codigo          INT          NOT NULL,
    SoPe_Empresa         VARCHAR(60)   NOT NULL,
    SoPe_Ciudad          VARCHAR(60)   NOT NULL,
    SoPe_Requiere        VARCHAR(75)   NOT NULL,
    SoPe_Fecha           DATE          NOT NULL,
    SoPe_Telefono        VARCHAR(15)   NOT NULL,
    SoPe_Firma           VARCHAR(60)   NOT NULL
);
```

```
DROP TABLE SAPW_Historia_Estudiante CASCADE;
CREATE TABLE SAPW_Historia_Estudiante(
    Estu_Matricula       VARCHAR(10)   NOT NULL,
    NiAc_Codigo          INT          NOT NULL,
    Carr_Codigo          INT          NOT NULL,
    Mate_Codigo          VARCHAR(10)   NOT NULL,
    Para_Paralelo        INT          NOT NULL,
    Para_Ano             INT          NOT NULL,
    Para_Termino         INT          NOT NULL,
    Hies_Nota            INT          NOT NULL,
    Esta_Codigo          INT          NOT NULL
);
```

DROP TABLE SAPW_Contacto CASCADE;

CREATE TABLE SAPW_Contacto (

Cont_Codigo	INT	NOT NULL,
Esta_Codigo	INT	NOT NULL,
Cont_Fecha	VARCHAR(20)	NOT NULL,
Cont_Nombre	VARCHAR(60)	NOT NULL,
Cont_Direccion	VARCHAR(60),	
Cont_Telefono	VARCHAR(10),	
Cont_Perfil	VARCHAR(20)	NOT NULL,
Cont_Edad	VARCHAR(20)	NOT NULL,
Cont_Sexo	VARCHAR(10)	NOT NULL,
Cont_Organizacion	VARCHAR(60)	NOT NULL,
Cont_Correo	VARCHAR(100)	NOT NULL,
Cont_Comentario	VARCHAR(10000)	NOT NULL,
Cont_DirigidoA	VARCHAR(100)	NOT NULL

);

DROP TABLE SAPW_Horario_Seminario CASCADE;

CREATE TABLE SAPW_Horario_Seminario (

HoSe_Numero	INT	NOT NULL,
Hora_Codigo	INT	NOT NULL,
Labo_Codigo	INT	NOT NULL,
Cert_Codigo	INT	NOT NULL,
Semi_Codigo	INT	NOT NULL,
Pers_Cedula	VARCHAR(10)	NOT NULL,
HoSe_CupoDisponible	INT	NOT NULL,
HoSe_Dias	VARCHAR(60)	NOT NULL,
HoSe_Observacion	VARCHAR(60)	NOT NULL

);

DROP TABLE SAPW_Observaciones CASCADE;

CREATE TABLE SAPW_Observaciones (

Obse_Codigo	INT	NOT NULL,
Obse_Descripcion	VARCHAR(10000)	NOT NULL

);

DROP TABLE SAPW_Seminario CASCADE;

CREATE TABLE SAPW_Seminario (

Semi_Codigo	INT	NOT NULL,
Cert_Codigo	INT	NOT NULL,
Semi_Nombre	VARCHAR(256)	NOT NULL,
Semi_Descripcion	VARCHAR(10000)	NOT NULL,
Semi_Duracion	INT	NOT NULL,
Semi_Costo	FLOAT	NOT NULL,
Semi_Cupo	INT	NOT NULL,
Semi_FechaInicio	DATE,	
Esta_Codigo	INT	NOT NULL,
Semi_Archivo	VARCHAR(300)	

);

DROP TABLE SAPW_Certificado CASCADE;

CREATE TABLE SAPW_Certificado (

Cert_Codigo	INT	NOT NULL,
Cert_Nombre	VARCHAR(256)	NOT NULL,
Cert_Dirigido	VARCHAR(10000)	NOT NULL,
Cert_Objeto	VARCHAR(10000)	NOT NULL,
Cert_Requerimiento	VARCHAR(256)	NOT NULL,
Cert_Lugar	VARCHAR(200)	

);

DROP TABLE SAPW_Documentos CASCADE;

CREATE TABLE SAPW_Documentos (

Docu_Linea	INT	NOT NULL,
Docu_Descripcion	VARCHAR(10000)	NOT NULL

);

DROP TABLE SAPW_Articulo CASCADE;

CREATE TABLE SAPW_Articulo(

Arti_Codigo	INT	NOT NULL,
Tema_Codigo	INT	NOT NULL,
Regl_Codigo	INT	NOT NULL,
Arti_Articulo	VARCHAR(10000)	NOT NULL

);

DROP TABLE SAPW_Tema CASCADE;

CREATE TABLE SAPW_Tema(

Tema_Codigo	INT	NOT NULL,
Regl_Codigo	INT	NOT NULL,
Tema_Titulo	VARCHAR(60)	NOT NULL

);

DROP TABLE SAPW_Reglamento CASCADE;

CREATE TABLE SAPW_Reglamento(

Regl_Codigo	INT	NOT NULL,
Regl_Titulo	VARCHAR(60)	NOT NULL,
Regl_ConsideracionGeneral	VARCHAR(10000)	NOT NULL

);

DROP TABLE SAPW_Integrante_Resolucion CASCADE;

CREATE TABLE SAPW_Integrante_Resolucion(

CaRe_Codigo	INT	NOT NULL,
Pers_Cedula	VARCHAR(10)	NOT NULL,
InRe_Cargo	VARCHAR(75)	NOT NULL

);

DROP TABLE SAPW_Detalle_Resolucion CASCADE;

CREATE TABLE SAPW_Detalle_Resolucion(

DeRe_Linea	INT	NOT NULL,
CaRe_Codigo	INT	NOT NULL,
DeRe_Descripcion	VARCHAR(10000)	NOT NULL);

```
DROP TABLE SAPW_Cabecera_Resolucion CASCADE;
CREATE TABLE SAPW_Cabecera_Resolucion(
    CaRe_Codigo      INT          NOT NULL,
    CaRe_Descripcion  VARCHAR(10000) NOT NULL,
    CaRe_Fecha        DATE         NOT NULL
);
```

```
DROP TABLE SAPW_Delegado CASCADE;
CREATE TABLE SAPW_Delegado (
    Dele_Numero      INT          NOT NULL,
    Dire_Numero      INT          NOT NULL,
    Divi_Codigo      INT          NOT NULL,
    Carg_Codigo      INT          NOT NULL,
    Dele_Cedula      VARCHAR(10)  NOT NULL,
    Dele_Nombre      VARCHAR(80)  NOT NULL
);
```

```
DROP TABLE SAPW_Directivo CASCADE;
CREATE TABLE SAPW_Directivo (
    Dire_Numero      INT          NOT NULL,
    Divi_Codigo      INT          NOT NULL,
    Carg_Codigo      INT          NOT NULL,
    Dire_Cedula      VARCHAR(10)  NOT NULL,
    Dire_NombreEncargado VARCHAR(80) NOT NULL
);
```

```
DROP TABLE SAPW_Division CASCADE;
CREATE TABLE SAPW_Division (
    Divi_Codigo      INT          NOT NULL,
    Divi_Nombre      VARCHAR(40)  NOT NULL,
    Divi_Descripcion  VARCHAR(10000) NOT NULL
);
```

```
DROP TABLE SAPW_Producto CASCADE;
CREATE TABLE SAPW_Producto (
    Prod_Numero      INT          NOT NULL,
    Serv_Numero      INT          NOT NULL,
    Depa_Codigo      INT          NOT NULL,
    Clie_Codigo      INT          NOT NULL,
    Prod_Nombre      VARCHAR(60)  NOT NULL,
    Prod_Descripcion  VARCHAR(10000) NOT NULL,
    Prod_Costo       FLOAT        NOT NULL,
    Prod_Comentario   VARCHAR(10000) NOT NULL,
    Prod_Fecha       DATE         NOT NULL,
    Prod_Url         VARCHAR(1000) NOT NULL,
    Prod_Visita      INT          NOT NULL,
    Foto_Codigo      INT
);
```


DROP TABLE SAPW_Servicio CASCADE;

CREATE TABLE SAPW_Servicio (

Serv_Numero	INT	NOT NULL,
Depa_Codigo	INT	NOT NULL,
Serv_Nombre	VARCHAR(60)	NOT NULL,
Serv_Descripcion	VARCHAR(10000)	NOT NULL,
Serv_Visita	INT	NOT NULL,
Serv_Url	VARCHAR(1000)	NOT NULL

);

DROP TABLE SAPW_Departamento CASCADE;

CREATE TABLE SAPW_Departamento (

Depa_Codigo	INT	NOT NULL,
Pers_Cedula	VARCHAR(10)	NOT NULL,
Depa_Nombre	VARCHAR(60)	NOT NULL,
Depa_Descripcion	VARCHAR(10000)	NOT NULL,
Depa_Visita	INT	NOT NULL,
Depa_Extension	INT	NOT NULL,
Depa_int_ext	INT	

);

DROP TABLE SAPW_Deber CASCADE;

CREATE TABLE SAPW_Deber (

Debe_Numero	INT	NOT NULL,
CaEs_Numero	INT	NOT NULL,
Para_Paralelo	INT	NOT NULL,
Anio_Codigo	INT	NOT NULL,
Peri_Codigo	INT	NOT NULL,
NiAc_Codigo	INT	NOT NULL,
Carr_Codigo	INT	NOT NULL,
Mate_Codigo	VARCHAR(10)	NOT NULL,
Debe_Descripcion	VARCHAR(10000)	NOT NULL,
Debe_FechaPublicacion	DATE	NOT NULL,
Debe_FechaEntrega	DATE	NOT NULL,
Debe_NombreArchivo	VARCHAR(100)	

);

DROP TABLE SAPW_Capitulo_Estudio CASCADE;

CREATE TABLE SAPW_Capitulo_Estudio (

CaEs_Numero	INT	NOT NULL,
Para_Paralelo	INT	NOT NULL,
Anio_Codigo	INT	NOT NULL,
Peri_Codigo	INT	NOT NULL,
NiAc_Codigo	INT	NOT NULL,
Carr_Codigo	INT	NOT NULL,
Mate_Codigo	VARCHAR(10)	NOT NULL,
CaEs_Titulo	VARCHAR(40)	NOT NULL,
CaEs_Descripcion	VARCHAR(10000)	NOT NULL,
CaEs_NombreArchivo	VARCHAR(100)	

);

DROP TABLE SAPW_Politica CASCADE;

CREATE TABLE SAPW_Politica (

Poli_Numero	INT	NOT NULL,
NiAc_Codigo	INT	NOT NULL,
Carr_Codigo	INT	NOT NULL,
Mate_Codigo	VARCHAR(10)	NOT NULL,
Para_Paralelo	INT	NOT NULL,
Anio_Codigo	INT	NOT NULL,
Peri_Codigo	INT	NOT NULL,
Poli_Descripcion	VARCHAR(10000)	NOT NULL

);

DROP TABLE SAPW_Desgloce_Trabajo CASCADE;

CREATE TABLE SAPW_Desgloce_Trabajo (

DeTr_Linea	INT	NOT NULL,
Desg_Linea	INT	NOT NULL,
Nota_Secuencial	INT	NOT NULL,
Estu_Matricula	VARCHAR(10)	NOT NULL,
NiAc_Codigo	INT	NOT NULL,
Carr_Codigo	INT	NOT NULL,
Mate_Codigo	VARCHAR(10)	NOT NULL,
Para_Paralelo	INT	NOT NULL,
Anio_Codigo	INT	NOT NULL,
Peri_Codigo	INT	NOT NULL,
DeTr_Nota	FLOAT	NOT NULL,
DeTr_Observacion	VARCHAR(10000)	NOT NULL

);

DROP TABLE SAPW_Desgloce CASCADE;

CREATE TABLE SAPW_Desgloce (

Desg_Linea	INT	NOT NULL,
Nota_Secuencial	INT	NOT NULL,
Estu_Matricula	VARCHAR(10)	NOT NULL,
NiAc_Codigo	INT	NOT NULL,
Carr_Codigo	INT	NOT NULL,
Mate_Codigo	VARCHAR(10)	NOT NULL,
Para_Paralelo	INT	NOT NULL,
Anio_Codigo	INT	NOT NULL,
Peri_Codigo	INT	NOT NULL,
Desg_Nota	FLOAT	NOT NULL,
Tipo_Codigo	INT	NOT NULL

);

DROP TABLE SAPW_Nota CASCADE;

CREATE TABLE SAPW_Nota (

Nota_Secuencial	INT	NOT NULL,
Estu_Matricula	VARCHAR(10)	NOT NULL,
NiAc_Codigo	INT	NOT NULL,
Carr_Codigo	INT	NOT NULL,
Mate_Codigo	VARCHAR(10)	NOT NULL,

Para_Paralelo	INT	NOT NULL,
Anio_Codigo	INT	NOT NULL,
Peri_Codigo	INT	NOT NULL,
Tipo_Codigo	INT	NOT NULL,
Nota_Nota	FLOAT	NOT NULL,
Nota_Observacion	VARCHAR(10000)	NOT NULL

);

DROP TABLE SAPW_Estudiante_Paralelo CASCADE;

CREATE TABLE SAPW_Estudiante_Paralelo (

Estu_Matricula	VARCHAR(10)	NOT NULL,
NiAc_Codigo	INT	NOT NULL,
Carr_Codigo	INT	NOT NULL,
Mate_Codigo	VARCHAR(10)	NOT NULL,
Para_Paralelo	INT	NOT NULL,
Anio_Codigo	INT	NOT NULL,
Peri_Codigo	INT	NOT NULL,
EsPa_Nota	FLOAT,	
Esta_Codigo	INT	NOT NULL

);

DROP TABLE SAPW_Paralelo CASCADE;

CREATE TABLE SAPW_Paralelo (

Anio_Codigo	INT	NOT NULL,
Peri_Codigo	INT	NOT NULL,
Para_Paralelo	INT	NOT NULL,
NiAc_Codigo	INT	NOT NULL,
Carr_Codigo	INT	NOT NULL,
Mate_Codigo	VARCHAR(10)	NOT NULL,
Pers_Cedula	VARCHAR(10)	NOT NULL

);

DROP TABLE SAPW_Materia_Nivel CASCADE;

CREATE TABLE SAPW_Materia_Nivel (

Mate_Codigo	VARCHAR(10)	NOT NULL,
NiAc_Codigo	INT	NOT NULL,
Carr_Codigo	INT	NOT NULL

);

DROP TABLE SAPW_Materia CASCADE;

CREATE TABLE SAPW_Materia (

Mate_Codigo	VARCHAR(10)	NOT NULL,
Tipo_Codigo	INT	NOT NULL,
Mate_Nombre	VARCHAR(40)	NOT NULL,
Mate_Descripcion	VARCHAR(10000)	NOT NULL

);

DROP TABLE SAPW_Nivel_Academico CASCADE;

CREATE TABLE SAPW_Nivel_Academico (

NiAc_Codigo	INT	NOT NULL,
-------------	-----	-----------

```
Carr_Codigo      INT      NOT NULL,
NiAc_Descripcion VARCHAR(10000) NOT NULL
);
```

```
DROP TABLE SAPW_Docente_Carrera CASCADE;
CREATE TABLE SAPW_Docente_Carrera (
    Pers_Cedula    VARCHAR(10)    NOT NULL,
    Carr_Codigo    INT            NOT NULL
);
```

```
DROP TABLE SAPW_Deuda_Estudiante CASCADE;
CREATE TABLE SAPW_Deuda_Estudiante (
    DeEs_Linea      INT            NOT NULL,
    Estu_Matricula  VARCHAR(10)    NOT NULL,
    Carr_Codigo     INT            NOT NULL,
    DeEs_Motivo     VARCHAR(10000) NOT NULL,
    DeEs_Fecha_Vencimiento DATE,
    DeEs_Valores    FLOAT          NOT NULL
);
```

```
DROP TABLE SAPW_Estudiante_Carrera CASCADE;
CREATE TABLE SAPW_Estudiante_Carrera (
    Estu_Matricula  VARCHAR(10)    NOT NULL,
    Carr_Codigo     INT            NOT NULL,
    Esta_Codigo     INT            NOT NULL
);
```

```
DROP TABLE SAPW_Promocion CASCADE;
CREATE TABLE SAPW_Promocion(
    Prom_Codigo     INT            NOT NULL,
    Anio_Codigo     INT            NOT NULL,
    Carr_Codigo     INT            NOT NULL,
    Estu_Matricula  VARCHAR(10),
    Prom_Nombre     VARCHAR(60)    NOT NULL,
    Prom_mail       VARCHAR (100),
    Prom_Promedio   FLOAT          NOT NULL,
    Prom_SitioWebUrl VARCHAR (100)
);
```

```
DROP TABLE SAPW_Periodo_Anio_Carrera CASCADE;
CREATE TABLE SAPW_Periodo_Anio_Carrera (
    Anio_Codigo     INT            NOT NULL,
    Carr_Codigo     INT            NOT NULL,
    Peri_Codigo     INT            NOT NULL,
    Peri_Descripcion VARCHAR(60)    NOT NULL,
    Esta_Codigo     INT            NOT NULL,
    Esta_Periodo    INT            NOT NULL
);
```

```
DROP TABLE SAPW_Anio_Carrera CASCADE;
CREATE TABLE SAPW_Anio_Carrera (
    Anio_Codigo      INT      NOT NULL,
    Carr_Codigo      INT      NOT NULL,
    Foto_Codigo      INT      NOT NULL,
    Esta_Codigo      INT      NOT NULL,
    Esta_Anio        INT      NOT NULL
);
```

```
DROP TABLE SAPW_Anio CASCADE;
CREATE TABLE SAPW_Anio (
    Anio_Codigo      INT NOT NULL,
    Anio_Descripcion  INT NOT NULL
);
```

```
DROP TABLE SAPW_Perfil_Empresarial CASCADE;
CREATE TABLE SAPW_Perfil_Empresarial(
    PeEm_Codigo      INT      NOT NULL,
    Carr_Codigo      INT      NOT NULL,
    Tipo_Codigo      INT      NOT NULL,
    PeEm_Descripcion  VARCHAR(10000) NOT NULL
);
```

```
DROP TABLE SAPW_Carrera CASCADE;
CREATE TABLE SAPW_Carrera (
    Carr_Codigo      INT      NOT NULL,
    Tipo_Codigo      INT      NOT NULL,
    Carr_Nombre      VARCHAR(60) NOT NULL,
    Carr_Descripcion  VARCHAR(10000) NOT NULL,
    Carr_Titulo      VARCHAR(60) NOT NULL,
    Carr_Duracion    VARCHAR(60) NOT NULL,
    Carr_Flujo       VARCHAR(1000),
    Carr_Pagina      VARCHAR(1000),
    Carr_Imagen      VARCHAR(20)
);
```

```
DROP TABLE SAPW_Rol_Sistema CASCADE;
CREATE TABLE SAPW_Rol_Sistema (
    GrUs_Codigo      INT      NOT NULL,
    OpSi_Codigo      INT      NOT NULL,
    OpSi_OpcionCodigo INT      NOT NULL,
    Esta_Codigo      INT      NOT NULL
);
```

```
DROP TABLE SAPW_Perfil_Usuario CASCADE;
CREATE TABLE SAPW_Perfil_Usuario (
    Usua_Login       VARCHAR(40) NOT NULL,
    OpSi_Codigo      INT      NOT NULL,
    OpSi_OpcionCodigo INT      NOT NULL,
    Esta_Codigo      INT      NOT NULL);
```

DROP TABLE SAPW_Opcion_Sistema CASCADE;

CREATE TABLE SAPW_Opcion_Sistema (

OpSi_Codigo	INT	NOT NULL,
OpSi_OpcionCodigo	INT	NOT NULL,
OpSi_NombreLogico	VARCHAR(60)	NOT NULL,
OpSi_NombreFisico	VARCHAR(60)	NOT NULL,
OpSi_Nivel	INT	NOT NULL

);

DROP TABLE SAPW_Rol_Web CASCADE;

CREATE TABLE SAPW_Rol_Web (

GrUs_Codigo	INT	NOT NULL,
OpWe_Codigo	INT	NOT NULL,
OpWe_OpcionCodigo	INT	NOT NULL,
Esta_Codigo	INT	NOT NULL

);

DROP TABLE SAPW_Perfil_Web CASCADE;

CREATE TABLE SAPW_Perfil_Web (

Usua_Login	VARCHAR(40)	NOT NULL,
OpWe_Codigo	INT	NOT NULL,
Esta_Codigo	INT	NOT NULL

);

DROP TABLE SAPW_Tip CASCADE;

CREATE TABLE SAPW_Tip (

Tip_Numero	INT	NOT NULL,
OpWe_Codigo	INT	NOT NULL,
Tip_Resumen	VARCHAR(60)	NOT NULL,
Tip_Descripcion	VARCHAR(10000)	NOT NULL,
Tip_Visita	INT	NOT NULL,
Tip_Publicar	VARCHAR(1)	NOT NULL

CHECK (Tip_Publicar = 'N' OR Tip_Publicar = 'A')

);

DROP TABLE SAPW_Opcion_Web CASCADE;

CREATE TABLE SAPW_Opcion_Web (

OpWe_Codigo	INT	NOT NULL,
OpWe_OpcionCodigo	INT	NOT NULL,
OpWe_Nombre	VARCHAR(60)	NOT NULL,
OpWe_Url	VARCHAR(1000),	
OpWe_UrlN	VARCHAR(1000),	
OpWe_UrlP	VARCHAR(1000),	
OpWe_UrlPN	VARCHAR(1000),	
OpWe_Visita	INT	NOT NULL,
OpWe_General	VARCHAR(1)	NOT NULL

CHECK (OpWe_General = 'S' OR OpWe_General = 'N' OR OpWe_General = 'E' OR OpWe_General = 'V'),

OpWe_Descripcion	VARCHAR(10000)	NOT NULL,
OpWe_DescripcionGeneral	VARCHAR(10000)	NOT NULL,

```

OpWe_Departamento    INT          NOT NULL,
Foto_Codigo          INT,
Esta_Codigo          INT          NOT NULL
);

```

DROP TABLE SAPW_Usuario CASCADE;

CREATE TABLE SAPW_Usuario (

```

Usua_Login           VARCHAR(40)   NOT NULL,
Esta_Codigo          INT          NOT NULL,
Usua_Propietario     VARCHAR(10)   NOT NULL,
GrUs_Codigo          INT          NOT NULL,
Usua_Contrasenia    VARCHAR(40)   NOT NULL,
Usua_FechaLimite     DATE,
Usua_TipoPropietario INT          NOT NULL,
Usua_Conexion        INT          NOT NULL,
Usua_ConexionSistema INT          NOT NULL,
Usua_InvitaChat      INT          NOT NULL
);

```

DROP TABLE SAPW_Grupo_Usuario CASCADE;

CREATE TABLE SAPW_Grupo_Usuario (

```

GrUs_Codigo          INT          NOT NULL,
Esta_Codigo          INT          NOT NULL,
GrUs_Descripcion     VARCHAR(60)   NOT NULL,
GrUs_Pertenece       INT          NOT NULL
CHECK (GrUs_Pertenece = 1 OR GrUs_Pertenece = 2),
      --1 ==> Sistema    --2 ==> Pagina
GrUs_Visita          INT          NOT NULL
);

```

DROP TABLE SAPW_Estudiante CASCADE;

CREATE TABLE SAPW_Estudiante (

```

Estu_Matricula       VARCHAR(10)   NOT NULL,
Esta_Codigo          INT          NOT NULL,
Estu_Cedula          VARCHAR(10)   NOT NULL,
Estu_Apellidos       VARCHAR(40)   NOT NULL,
Estu_Nombres         VARCHAR(40)   NOT NULL,
Estu_FechaNacimiento DATE          NOT NULL,
Estu_Direccion       VARCHAR(40)   NOT NULL,
Estu_Telefono        VARCHAR(9)    NOT NULL,
Estu_Email           VARCHAR(100)   NOT NULL,
Estu_EstadoCivil     VARCHAR(1)     NOT NULL
CHECK (Estu_EstadoCivil = 'S' OR Estu_EstadoCivil = 'C' OR
      Estu_EstadoCivil = 'V' OR Estu_EstadoCivil = 'D' OR
      Estu_EstadoCivil = 'U'),
Estu_Sexo            VARCHAR(1)     NOT NULL
CHECK (Estu_Sexo = 'M' OR Estu_Sexo = 'F'),
Tipo_Identificacion  INT          NOT NULL
);

```


DROP TABLE SAPW_Persona CASCADE;

CREATE TABLE SAPW_Persona (

Pers_Cedula	VARCHAR(10)	NOT NULL,
TiPr_Codigo	INT	NOT NULL,
Tipo_Codigo	INT	NOT NULL,
Pers_Apellidos	VARCHAR(40)	NOT NULL,
Pers_Nombres	VARCHAR(40)	NOT NULL,
Pers_FechaNacimiento	DATE	NOT NULL,
Pers_Direccion	VARCHAR(40)	NOT NULL,
Pers_Telefono	VARCHAR(9)	NOT NULL,
Pers_Email	VARCHAR(100)	NOT NULL,
Pers_EstadoCivil	VARCHAR(1)	NOT NULL
CHECK (Pers_EstadoCivil = 'S' OR Pers_EstadoCivil = 'C' OR Pers_EstadoCivil = 'V' OR Pers_EstadoCivil = 'D' OR Pers_EstadoCivil = 'U'),		
Pers_Sexo	VARCHAR(1)	NOT NULL
CHECK (Pers_Sexo = 'M' OR Pers_Sexo = 'F'),		
Foto_Codigo	INT,	
Pers_Tipo	INT	NOT NULL,
Tipo_Identificacion	INT	NOT NULL,
Esta_Codigo	INT	NOT NULL,
Pers_UrlSitio	VARCHAR(100)	

);

DROP TABLE SAPW_Titulo_Profesional CASCADE;

CREATE TABLE SAPW_Titulo_Profesional (

TiPr_Codigo	INT	NOT NULL,
TiPr_Nombre	VARCHAR(60)	NOT NULL,
TiPr_Descripcion	VARCHAR(10000)	NOT NULL,
TiPr_Abreviatura	VARCHAR(10)	NOT NULL

);

DROP TABLE SAPW_Cargo CASCADE;

CREATE TABLE SAPW_Cargo (

Carg_Codigo	INT	NOT NULL,
Carg_Descripcion	VARCHAR(60)	NOT NULL

);

DROP TABLE SAPW_Horario CASCADE;

CREATE TABLE SAPW_Horario(

Hora_Codigo	INT	NOT NULL,
Hora_Inicio	TIME	NOT NULL,
Hora_Fin	TIME	NOT NULL

);

DROP TABLE SAPW_Link CASCADE;

CREATE TABLE SAPW_Link (

Link_Codigo	INT	NOT NULL,
Tipo_Codigo	INT	NOT NULL,
Link_Topico	VARCHAR(60)	NOT NULL,


```

Link_Descripcion  VARCHAR(10000) NOT NULL,
Link_Url          VARCHAR(1000)  NOT NULL,
Link_Visita       INT             NOT NULL,
Link_Tipo         VARCHAR(1)     NOT NULL
CHECK (Link_Tipo = 'F' OR Link_Tipo = 'A' OR
      Link_Tipo = 'V' OR Link_Tipo = 'E' OR
      Link_Tipo = 'N'),
Link_Campus      VARCHAR(1)     NOT NULL
CHECK (Link_Campus = 'S' OR Link_Campus = 'N')
);

```

DROP TABLE SAPW_Cliente CASCADE;

```

CREATE TABLE SAPW_Cliente (
  Clie_Codigo      INT             NOT NULL,
  Tipo_Codigo      INT             NOT NULL,
  Clie_Nombre      VARCHAR(40)     NOT NULL,
  Clie_Direccion   VARCHAR(40)     NOT NULL,
  Clie_Telefono    VARCHAR(9)      NOT NULL,
  Clie_Email       VARCHAR(100)    NOT NULL,
  Clie_Url         VARCHAR(100)    NOT NULL
);

```

DROP TABLE SAPW_Tipo CASCADE;

```

CREATE TABLE SAPW_Tipo (
  Tipo_Codigo      INT             NOT NULL,
  Tipo_Descripcion VARCHAR(60)     NOT NULL,
  Tipo_Clasificacion INT          NOT NULL
);

```

DROP TABLE SAPW_Salon_Clase CASCADE;

```

CREATE TABLE SAPW_Salon_Clase(
  SaCl_Codigo      INT             NOT NULL,
  SaCl_Descripcion VARCHAR(60)     NOT NULL
);

```

DROP TABLE SAPW_Laboratorio CASCADE;

```

CREATE TABLE SAPW_Laboratorio(
  Labo_Codigo      INT             NOT NULL,
  Labo_Descripcion VARCHAR(60)     NOT NULL,
  Labo_Ubicacion   VARCHAR(60)     NOT NULL,
  Labo_Cupo       INT             NOT NULL
);

```

DROP TABLE SAPW_Foto CASCADE;

```

CREATE TABLE SAPW_Foto(
  Foto_Codigo      INT             NOT NULL,
  Foto_Numero      INT             NOT NULL,
  Foto_Direccion   VARCHAR(100),
  Foto_Thumbnails  VARCHAR(100),
  Foto_Ascii       VARCHAR(100),

```



```
Foto_Descripcion    VARCHAR(256),
Foto_Estado         INT          NOT NULL
);

DROP TABLE SAPW_Direccion_Remota CASCADE;
CREATE TABLE SAPW_Direccion_Remota(
    DiRe_IpRemota     VARCHAR(20)    NOT NULL,
    DiRe_PuertoRemoto  VARCHAR(20)    NOT NULL,
    DiRe_FechaHoraVisita VARCHAR(20)    NOT NULL
);

DROP TABLE SAPW_Capacitacion_In_Situ CASCADE;
CREATE TABLE SAPW_Capacitacion_In_Situ(
    CaIs_Codigo        INT          NOT NULL,
    CaIs_NombreEmpresa VARCHAR(60)   NOT NULL,
    CaIs_PersonaResponsable VARCHAR(60) NOT NULL,
    CaIs_TelefonoFax    VARCHAR(12)   NOT NULL,
    CaIs_Email          VARCHAR(100)   NOT NULL,
    CaIs_SeminarioRequerido VARCHAR(10000) NOT NULL,
    CaIs_Num_Participantes INT        NOT NULL,
    CaIs_HorarioClases  VARCHAR(80)   NOT NULL,
    CaIs_FechaInicio    VARCHAR(80)   NOT NULL,
    Esta_Codigo         INT          NOT NULL,
    Pers_Cedula         VARCHAR(10)   NOT NULL
);

DROP TABLE SAPW_Estado CASCADE;
CREATE TABLE SAPW_Estado (
    Esta_Codigo        INT          NOT NULL,
    Esta_Descripcion   VARCHAR(60)   NOT NULL,
    Esta_Clasificacion INT          NOT NULL
);

DROP TABLE SAPW_BackupOnLine CASCADE;
CREATE TABLE SAPW_BackupOnLine (
    BaOn_Numero        INT          NOT NULL,
    Usua_Login         VARCHAR(40)   NOT NULL,
    BaOn_Archivo        VARCHAR(10000) NOT NULL,
    BaOn_Fecha         TIMESTAMP NOT NULL
);

DROP TABLE SAPW_Detalle_Testimonios CASCADE;
CREATE TABLE SAPW_Detalle_Testimonios (
    Test_Numero        INT          NOT NULL,
    Preg_Numero        INT          NOT NULL,
    Det_Respuesta       VARCHAR(10000),
    Esta_Codigo         INT          NOT NULL
);
```

```
DROP TABLE SAPW_Preguntas_Testimonios CASCADE;
CREATE TABLE SAPW_Preguntas_Testimonios (
    Preg_Numero          INT          NOT NULL,
    Preg_Pregunta        VARCHAR(200) NOT NULL,
    Esta_Codigo          INT);
```

```
DROP TABLE SAPW_Testimonios CASCADE;
CREATE TABLE sSAPW_Testimonios (
    Test_Numero          INT          NOT NULL,
    Test_Nombre          VARCHAR(80)  NOT NULL,
    Test_Titulo          VARCHAR(80)  NOT NULL,
    Test_Cargo           VARCHAR(80)  NOT NULL,
    Test_Comentario      VARCHAR(200) NOT NULL,
    Foto_Codigo          INT,
    Test_Empresa         VARCHAR(80),
    Foto_Logo            INT,
    Test_Tiempo          VARCHAR(20),
    Esta_Codigo          INT,
    Foto_1               INT,
    Foto_2               INT,
    Foto_3               INT
);
```

```
DROP TABLE SAPW_Detalle_Fondo_Beca CASCADE;
CREATE TABLE SAPW_Detalle_Fondo_Beca(
    DeBe_Codigo          INT,
    FoBe_Codigo          INT,
    DeBe_Cobertura       VARCHAR(1000),
    DeBe_Beneficiario    VARCHAR(1000),
    DeBe_Beca            VARCHAR(1000),
    Esta_Codigo          INT
);
```

```
DROP TABLE SAPW_Fondo_Beca CASCADE;
CREATE TABLE SAPW_Fondo_Beca (
    FoBe_Codigo          INT          NOT NULL,
    FoBe_Titulo          VARCHAR(100),
    FoBe_Descripcion     VARCHAR(1000),
    FoBe_Objeto          VARCHAR(10000),
);
```

```
DROP TABLE SAPW_Materia_Req_Pasantia CASCADE;
CREATE TABLE SAPW_Materia_Req_Pasantia (
    MaPa_Codigo          VARCHAR(10) NOT NULL
);
```

```
DROP TABLE SAPW_Politica_Pasantia CASCADE;
CREATE TABLE SAPW_Politica_Pasantia (
    PoPa_Codigo          INT          NOT NULL,
    Carr_Codigo          INT,
```

```
PoPa_Descripcion    VARCHAR(100000),
Esta_Codigo         INT,
PoPa_Titulo         VARCHAR(1000),
);

DROP TABLE SAPW_Requisito_Beca CASCADE;
CREATE TABLE SAPW_Requisito_Beca (
    ReBe_Codigo       INT          NOT NULL,
    FoBe_Codigo       INT,
    ReBe_Descripcion  VARCHAR(10000)
);

DROP TABLE SAPW_Requisito_Pasantia CASCADE;
CREATE TABLE SAPW_Requisito_Pasantia (
    RePa_Codigo       INT          NOT NULL,
    RePa_Padre        INT,
    Carr_Codigo       INT,
    RePa_Descripcion  VARCHAR(10000),
    Esta_Codigo       INT,
    Repa_Nivel        INT,
);
```

4.1.2 Creación de Claves Primarias

```
ALTER TABLE SAPW_Tipo ADD CONSTRAINT Tipo_Codigo_Pk
    PRIMARY KEY (Tipo_Codigo);

ALTER TABLE SAPW_Titulo_Profesional ADD CONSTRAINT TiPr_Codigo_Pk
    PRIMARY KEY (TiPr_Codigo);

ALTER TABLE SAPW_Cargo ADD CONSTRAINT Carg_Codigo_Pk
    PRIMARY KEY (Carg_Codigo);

ALTER TABLE SAPW_Departamento ADD CONSTRAINT Depa_Codigo_Pk
    PRIMARY KEY (Depa_Codigo);

ALTER TABLE SAPW_Estado ADD CONSTRAINT Esta_Codigo_Pk
    PRIMARY KEY (Esta_Codigo);

ALTER TABLE SAPW_Forum ADD CONSTRAINT Foru_Codigo_Pk
    PRIMARY KEY (Foru_Numero);

ALTER TABLE SAPW_Recurso_Humano ADD CONSTRAINT ReHu_Codigo_Pk
    PRIMARY KEY (ReHu_Codigo);

ALTER TABLE SAPW_Solicitud_Personal ADD CONSTRAINT SoPe_Codigo_Pk
    PRIMARY KEY (SoPe_Codigo);
```

ALTER TABLE SAPW_Catalogo_Protcom ADD CONSTRAINT CaPr_Codigo_Pk
PRIMARY KEY (CaPr_Codigo);

ALTER TABLE SAPW_Division ADD CONSTRAINT Divi_Codigo_Pk
PRIMARY KEY (Divi_Codigo);

ALTER TABLE SAPW_Laboratorio ADD CONSTRAINT Labo_Codigo_Pk
PRIMARY KEY (Labo_Codigo);

ALTER TABLE SAPW_Horario ADD CONSTRAINT Hora_Codigo_Pk
PRIMARY KEY (Hora_Codigo);

ALTER TABLE SAPW_Salon_Clase ADD CONSTRAINT SaCl_Codigo_Pk
PRIMARY KEY (SaCl_Codigo);

ALTER TABLE SAPW_Certificado ADD CONSTRAINT Cert_Codigo_Pk
PRIMARY KEY (Cert_Codigo);

ALTER TABLE SAPW_Estado_Animo ADD CONSTRAINT EsAn_Codigo_Pk
PRIMARY KEY (EsAn_Codigo);

ALTER TABLE SAPW_Reglamento ADD CONSTRAINT Regl_Codigo_Pk
PRIMARY KEY (Regl_Codigo);

ALTER TABLE SAPW_Link ADD CONSTRAINT Link_Codigo_Pk
PRIMARY KEY (Link_Codigo);

ALTER TABLE SAPW_Persona ADD CONSTRAINT Pers_Cedula_Pk
PRIMARY KEY (Pers_Cedula);

ALTER TABLE SAPW_Politica_Reservacion ADD CONSTRAINT PoRe_Codigo_Pk
PRIMARY KEY (PoRe_Codigo);

ALTER TABLE SAPW_Documentos ADD CONSTRAINT Docu_Codigo_Pk
PRIMARY KEY (Docu_Linea);

ALTER TABLE SAPW_Anuncio ADD CONSTRAINT Anun_Codigo_Pk
PRIMARY KEY (Anun_Codigo);

ALTER TABLE SAPW_Sonido ADD CONSTRAINT Soni_Codigo_Pk
PRIMARY KEY (Soni_Codigo,EsAn_Codigo);

ALTER TABLE SAPW_Colores_Animo ADD CONSTRAINT CoAn_Codigo_Pk
PRIMARY KEY (CoAn_Codigo,EsAn_Codigo);

ALTER TABLE SAPW_Directivo ADD CONSTRAINT Dire_Codigo_Pk
PRIMARY KEY (Dire_Numero,Divi_Codigo);

ALTER TABLE SAPW_Tema ADD CONSTRAINT Tema_Codigo_Pk
PRIMARY KEY (Tema_Codigo,Regl_Codigo);

```
ALTER TABLE SAPW_Articulo ADD CONSTRAINT Arti_Codigo_Pk
    PRIMARY KEY (Arti_Codigo,Tema_Codigo,Regl_Codigo);

ALTER TABLE SAPW_Part es_Pagina ADD CONSTRAINT PaPa_Codigo_Pk
    PRIMARY KEY (PaPa_Codigo,CoAn_Codigo,EsAn_Codigo);

ALTER TABLE SAPW_Usuario ADD CONSTRAINT Usua_Login_Pk
    PRIMARY KEY (Usua_Login);

ALTER TABLE SAPW_Perfil_Empresarial ADD CONSTRAINT PeEm_Codigo_Pk
    PRIMARY KEY (PeEm_Codigo);

ALTER TABLE SAPW_Usuario_Forum ADD CONSTRAINT UsFo_LoNu_Pk
    PRIMARY KEY (Usua_Login, Foru_Numero);

ALTER TABLE SAPW_Contenido ADD CONSTRAINT Cont_Codigo_Pk
    PRIMARY KEY (Cont_Codigo);

ALTER TABLE SAPW_Empresa ADD CONSTRAINT Emp_Cedula_Pk
    PRIMARY KEY (Pers_Cedula);

ALTER TABLE SAPW_Respuesta ADD CONSTRAINT Resp_Numero_Pk
    PRIMARY KEY (Resp_Numero, Tema_Numero, Clas_Numero,
Foru_Numero);

ALTER TABLE SAPW_Clasificacion ADD CONSTRAINT Clas_Numero_Pk
    PRIMARY KEY (Clas_Numero, Foru_Numero);

ALTER TABLE SAPW_TemaForum ADD CONSTRAINT Tema_Numero_Pk
    PRIMARY KEY (Tema_Numero, Clas_Numero, Foru_Numero);

ALTER TABLE SAPW_Codigo_Fuente ADD CONSTRAINT CoFu_Codigo_Pk
    PRIMARY KEY (CoFu_Codigo);

ALTER TABLE SAPW_Promocion ADD CONSTRAINT Prom_Codigo_Pk
    PRIMARY KEY (Prom_Codigo);

ALTER TABLE SAPW_Evento ADD CONSTRAINT Even_Codigo_Pk
    PRIMARY KEY (Even_Codigo);

ALTER TABLE SAPW_Foto ADD CONSTRAINT Foto_Numero_Pk
    PRIMARY KEY (Foto_Codigo, Foto_Numero);

ALTER TABLE SAPW_Cabecera_Auditoria ADD CONSTRAINT CaAu_Codigo_Pk
    PRIMARY KEY (CaAu_Codigo);

ALTER TABLE SAPW_Detalle_Auditoria ADD CONSTRAINT DeAu_Codigo_Pk
    PRIMARY KEY (DeAu_Linea, CaAu_Codigo);
```



```
ALTER TABLE SAPW_Reservacion_Laboratorio ADD CONSTRAINT
ReLa_Codigo_Pk
    PRIMARY KEY (ReLa_Numero);

ALTER TABLE SAPW_Solicitud_Pasantia ADD CONSTRAINT SoPa_Codigo_Pk
    PRIMARY KEY (SoPa_Codigo);

ALTER TABLE SAPW_Pasantia ADD CONSTRAINT Pasa_Codigo_Pk
    PRIMARY KEY (Sopa_Codigo);

ALTER TABLE SAPW_Opcion_Sistema ADD CONSTRAINT OpSi_Codigo_Pk
    PRIMARY KEY (OpSi_Codigo, OpSi_OpcionCodigo);

ALTER TABLE SAPW_Perfil_Usuario ADD CONSTRAINT PerF_Login_Pk
    PRIMARY KEY (Usua_Login, OpSi_Codigo, OpSi_OpcionCodigo);

ALTER TABLE SAPW_Grupo_Usuario ADD CONSTRAINT GrUs_Codigo_Pk
    PRIMARY KEY (GrUs_Codigo);

ALTER TABLE SAPW_Rol_Sistema ADD CONSTRAINT RoSi_Codigo_Pk
    PRIMARY KEY (GrUs_Codigo, OpSi_Codigo, OpSi_OpcionCodigo);

ALTER TABLE SAPW_Opcion_Web ADD CONSTRAINT OpWe_Codigo_Pk
    PRIMARY KEY (OpWe_Codigo);

ALTER TABLE SAPW_Rol_Web ADD CONSTRAINT RoWe_Codigo_Pk
    PRIMARY KEY (GrUs_Codigo, OpWe_Codigo, OpWe_OpcionCodigo);

ALTER TABLE SAPW_Perfil_Web ADD CONSTRAINT PeWe_Login_Pk
    PRIMARY KEY (Usua_Login, OpWe_Codigo);

ALTER TABLE SAPW_Servicio ADD CONSTRAINT Serv_Numero_Pk
    PRIMARY KEY (Serv_Numero, Depa_Codigo);

ALTER TABLE SAPW_Cliente ADD CONSTRAINT Clie_Codigo_Pk
    PRIMARY KEY (Clie_Codigo);

ALTER TABLE SAPW_Producto ADD CONSTRAINT Prod_Numero_Pk
    PRIMARY KEY (Prod_Numero, Serv_Numero, Depa_Codigo);

ALTER TABLE SAPW_Tip ADD CONSTRAINT Tip_Codigo_Pk
    PRIMARY KEY (Tip_Numero);

ALTER TABLE SAPW_Contacto ADD CONSTRAINT Contac_Codigo_Pk
    PRIMARY KEY (Cont_Codigo);

ALTER TABLE SAPW_Buzon ADD CONSTRAINT BuSu_Codigo_Pk
    PRIMARY KEY (BuSu_Codigo);
```

```
ALTER TABLE SAPW_Carrera ADD CONSTRAINT Carr_Codigo_Pk  
PRIMARY KEY (Carr_Codigo);
```

```
ALTER TABLE SAPW_Seminario ADD CONSTRAINT Semi_Codigo_Pk  
PRIMARY KEY (Semi_Codigo, Cert_Codigo);
```

```
ALTER TABLE SAPW_Observaciones ADD CONSTRAINT Obse_Codigo_Pk  
PRIMARY KEY (Obse_Codigo);
```

```
ALTER TABLE SAPW_Horario_Seminario ADD CONSTRAINT HoSe_Codigo_Pk  
PRIMARY KEY (HoSe_Numero);
```

```
ALTER TABLE SAPW_Materia ADD CONSTRAINT Mate_Codigo_Pk  
PRIMARY KEY (Mate_Codigo);
```

```
ALTER TABLE SAPW_Nivel_Academico ADD CONSTRAINT NiAc_Codigo_Pk  
PRIMARY KEY (NiAc_Codigo, Carr_Codigo);
```

```
ALTER TABLE SAPW_Docente_Carrera ADD CONSTRAINT DoCa_Codigo_Pk  
PRIMARY KEY (Pers_Cedula, Carr_Codigo);
```

```
ALTER TABLE SAPW_Materia_Nivel ADD CONSTRAINT MaNi_Codigo_Pk  
PRIMARY KEY (Mate_Codigo, NiAc_Codigo, Carr_Codigo);
```

```
ALTER TABLE SAPW_Paralelo ADD CONSTRAINT Para_Codigo_Pk  
PRIMARY KEY (Anio_Codigo, Peri_Codigo, Para_Paralelo, NiAc_Codigo,  
Carr_Codigo, Mate_Codigo);
```

```
ALTER TABLE SAPW_Horario_Paralelo ADD CONSTRAINT HoPa_Codigo_Pk  
PRIMARY KEY (Hora_Codigo, NiAc_Codigo, Carr_Codigo, Mate_Codigo,  
Para_Paralelo, Anio_Codigo, Peri_Codigo, HoPa_Dia);
```

```
ALTER TABLE SAPW_Politica ADD CONSTRAINT Poli_Codigo_Pk  
PRIMARY KEY (Poli_Numero, NiAc_Codigo, Carr_Codigo, Mate_Codigo,  
Para_Paralelo, Anio_Codigo, Peri_Codigo);
```

```
ALTER TABLE SAPW_Capitulo_Estudio ADD CONSTRAINT CaEs_Codigo_Pk  
PRIMARY KEY (CaEs_Numero, NiAc_Codigo, Carr_Codigo, Mate_Codigo,  
Para_Paralelo, Anio_Codigo, Peri_Codigo);
```

```
ALTER TABLE SAPW_Deber ADD CONSTRAINT Debe_Codigo_Pk  
PRIMARY KEY (Debe_Numero, NiAc_Codigo, Carr_Codigo, CaEs_Numero,  
Mate_Codigo, Para_Paralelo, Anio_Codigo, Peri_Codigo);
```

```
ALTER TABLE SAPW_Delegado ADD CONSTRAINT Dele_Codigo_Pk  
PRIMARY KEY (Dele_Numero, Dire_Numero, Divi_Codigo);
```

```
ALTER TABLE SAPW_Cabecera_Resolucion ADD CONSTRAINT CaRe_Codigo_Pk  
PRIMARY KEY (CaRe_Codigo);
```


ALTER TABLE SAPW_Detalle_Resolucion ADD CONSTRAINT DeRe_Codigo_Pk
PRIMARY KEY (DeRe_Linea, CaRe_Codigo);

ALTER TABLE SAPW_Integrante_Resolucion ADD CONSTRAINT
InRe_Codigo_Pk
PRIMARY KEY (CaRe_Codigo, Pers_Cedula);

ALTER TABLE SAPW_Estudiante ADD CONSTRAINT Estu_Codigo_Pk
PRIMARY KEY (Estu_Matricula);

ALTER TABLE SAPW_Estudiante_Paralelo ADD CONSTRAINT EsPa_Codigo_Pk
PRIMARY KEY (Estu_Matricula, NiAc_Codigo, Carr_Codigo, Mate_Codigo,
Para_Paralelo, Anio_Codigo, Peri_Codigo);

ALTER TABLE SAPW_Nota ADD CONSTRAINT Nota_Codigo_Pk
PRIMARY KEY (Nota_Secuencial, Estu_Matricula, NiAc_Codigo,
Carr_Codigo, Mate_Codigo, Para_Paralelo, Anio_Codigo, Peri_Codigo);

ALTER TABLE SAPW_Desgloce ADD CONSTRAINT Desg_Codigo_Pk
PRIMARY KEY (Desg_Linea, Nota_Secuencial, Estu_Matricula,
NiAc_Codigo, Carr_Codigo, Mate_Codigo, Para_Paralelo, Anio_Codigo,
Peri_Codigo);

ALTER TABLE SAPW_Historia_Estudiante ADD CONSTRAINT HiEs_Codigo_Pk
PRIMARY KEY (Estu_Matricula, NiAc_Codigo, Carr_Codigo, Mate_Codigo,
Para_Paralelo, Para_Ano, Para_Termino);

ALTER TABLE SAPW_Curriculum_Vitae ADD CONSTRAINT
CuVi_Pers_Cedula_Pk
PRIMARY KEY (Pers_Cedula);

ALTER TABLE SAPW_Diccionario ADD CONSTRAINT Dicc_Codigo_Pk
PRIMARY KEY (Dicc_Codigo);

ALTER TABLE SAPW_Deuda_Estudiante ADD CONSTRAINT
DeEs_CodigoDeuda_Pk
PRIMARY KEY (DeEs_Linea, Estu_Matricula, Carr_Codigo);

ALTER TABLE SAPW_Anio ADD CONSTRAINT Anio_Codigo_Pk
PRIMARY KEY (Anio_Codigo);

ALTER TABLE SAPW_Anio_Carrera ADD CONSTRAINT AnCa_Codigo_Pk
PRIMARY KEY (Carr_Codigo, Anio_Codigo);

ALTER TABLE SAPW_Periodo_Anio_Carrera ADD CONSTRAINT
Peri_Anio_Codigo_Pk
PRIMARY KEY (Anio_Codigo, Carr_Codigo, Peri_Codigo);

ALTER TABLE SAPW_Estudiante_Carrera ADD CONSTRAINT EsCa_Codigo_Pk
PRIMARY KEY (Estu_Matricula, Carr_Codigo);

```
ALTER TABLE SAPW_Desgloce_Trabajo ADD CONSTRAINT DeTr_Codigo_Pk  
PRIMARY KEY (DeTr_Linea, Desg_Linea, Nota_Secuencial, Estu_Matricula,  
NiAc_Codigo, Carr_Codigo, Mate_Codigo, Para_Paralelo, Anio_Codigo,  
Peri_Codigo);
```

```
ALTER TABLE SAPW_Capacitacion_In_Situ ADD CONSTRAINT CaIs_Codigo_Pk  
PRIMARY KEY (CaIs_Codigo);
```

```
ALTER TABLE SAPW_BackupOnLine ADD CONSTRAINT BaOn_Codigo_Pk  
PRIMARY KEY (BaOn_Numero);
```

```
ALTER TABLE SAPW_Testimonios ADD CONSTRAINT Test_Numero_Pk  
PRIMARY KEY (Test_Numero);
```

```
ALTER TABLE SAPW_detalle_testimonios ADD CONSTRAINT DeTe_Numero_Pk  
PRIMARY KEY (test_numero, preg_numero),
```

```
ALTER TABLE SAPW_preguntas_testimonios ADD CONSTRAINT  
Preg_Numero_Pk  
PRIMARY KEY (Preg_Numero);
```

```
ALTER TABLE SAPW_materia_req_pasantia ADD CONSTRAINT  
MaPa_Codigo_Pk  
PRIMARY KEY (mapa_codigo),
```

```
ALTER TABLE SAPW_politica_pasantia ADD CONSTRAINT PoPa_Codigo_Pk  
PRIMARY KEY (popa_codigo),
```

```
ALTER TABLE SAPW_requisito_pasantia ADD CONSTRAINT RePa_Codigo_Pk  
PRIMARY KEY (repa_codigo),
```

```
ALTER TABLE SAPW_requisito_beca ADD CONSTRAINT ReBe_Codigo_Pk  
PRIMARY KEY (rebe_codigo),
```

```
ALTER TABLE SAPW_fondo_beca ADD CONSTRAINT FoBe_Codigo_Pk  
PRIMARY KEY (Fobe_codigo);
```

```
ALTER TABLE SAPW_Pasantias_Empresa ADD CONSTRAINT Pasem_Codigo_Pk  
PRIMARY KEY (Pasem_Codigo);
```

4.1.3 Creación de Claves Secundarias

```
ALTER TABLE SAPW_Link ADD CONSTRAINT Link_Tipo_Codigo_Fk  
FOREIGN KEY (Tipo_Codigo)  
REFERENCES SAPW_Tipo(Tipo_Codigo);
```

```
ALTER TABLE SAPW_Persona ADD CONSTRAINT Pers_Tipo_Codigo_Fk  
FOREIGN KEY (Tipo_Codigo)  
REFERENCES SAPW_Tipo(Tipo_Codigo);
```

```
ALTER TABLE SAPW_Persona ADD CONSTRAINT Pers_Tipo_Ide_Fk  
    FOREIGN KEY (Tipo_Identificacion)  
    REFERENCES SAPW_Tipo(Tipo_Codigo);
```

```
ALTER TABLE SAPW_Persona ADD CONSTRAINT Pers_Tipo_Persona_Fk  
    FOREIGN KEY (Pers_Tipo)  
    REFERENCES SAPW_Tipo(Tipo_Codigo);
```

```
ALTER TABLE SAPW_Estudiente ADD CONSTRAINT Estu_Tipo_Ide_Fk  
    FOREIGN KEY (Tipo_Identificacion)  
    REFERENCES SAPW_Tipo(Tipo_Codigo);
```

```
ALTER TABLE SAPW_Perfil_Empresarial ADD CONSTRAINT  
    PerEmp_Tipo_Codigo_Fk  
    FOREIGN KEY (Tipo_Codigo)  
    REFERENCES SAPW_Tipo(Tipo_Codigo);
```

```
ALTER TABLE SAPW_Cliente ADD CONSTRAINT Clie_Tipo_Codigo_Fk  
    FOREIGN KEY (Tipo_Codigo)  
    REFERENCES SAPW_Tipo(Tipo_Codigo);
```

```
ALTER TABLE SAPW_Contenido ADD CONSTRAINT Conte_Tipo_Codigo_Fk  
    FOREIGN KEY (Tipo_Codigo)  
    REFERENCES SAPW_Tipo(Tipo_Codigo);
```

```
ALTER TABLE SAPW_Materia ADD CONSTRAINT Mate_TiPo_Codigo_Fk  
    FOREIGN KEY (TiPo_Codigo)  
    REFERENCES SAPW_Tipo(Tipo_Codigo);
```

```
ALTER TABLE SAPW_Persona ADD CONSTRAINT Pers_TiPr_Codigo_Fk  
    FOREIGN KEY (TiPr_Codigo)  
    REFERENCES SAPW_Titulo_Profesional(TiPr_Codigo);
```

```
ALTER TABLE SAPW_Contacto ADD CONSTRAINT Contac_Esta_Codigo_Fk  
    FOREIGN KEY (Esta_Codigo)  
    REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Buzon ADD CONSTRAINT BuSu_Pers_Cedula_Fk  
    FOREIGN KEY (Pers_Cedula)  
    REFERENCES SAPW_Persona(Pers_Cedula);
```

```
ALTER TABLE SAPW_Buzon ADD CONSTRAINT BuSu_Esta_Codigo_Fk  
    FOREIGN KEY (Esta_Codigo)  
    REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Buzon ADD CONSTRAINT BuSu_Esta_Buzon_Fk  
    FOREIGN KEY (Esta_Buzon)  
    REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Departamento ADD CONSTRAINT Depa_Pers_Cedula_Fk  
    FOREIGN KEY (Pers_Cedula)  
    REFERENCES SAPW_Persona(Pers_Cedula);
```

```
ALTER TABLE SAPW_Paralelo ADD CONSTRAINT Para_Pers_Cedula_Fk  
    FOREIGN KEY (Pers_Cedula)  
    REFERENCES SAPW_Persona(Pers_Cedula);
```

```
ALTER TABLE SAPW_Docente_Carrera ADD CONSTRAINT  
DoCa_Pers_Cedula_Fk  
    FOREIGN KEY (Pers_Cedula)  
    REFERENCES SAPW_Persona(Pers_Cedula);
```

```
ALTER TABLE SAPW_Horario_Seminario ADD CONSTRAINT  
HoSe_Pers_Cedula_Fk  
    FOREIGN KEY (Pers_Cedula)  
    REFERENCES SAPW_Persona(Pers_Cedula);
```

```
ALTER TABLE SAPW_Horario_Seminario ADD CONSTRAINT  
HoSe_Labo_Codigo_Fk  
    FOREIGN KEY (Labo_Codigo)  
    REFERENCES SAPW_Laboratorio(Labo_Codigo);
```

```
ALTER TABLE SAPW_Integrante_Resolucion ADD CONSTRAINT  
InRe_Pers_Cedula_Fk  
    FOREIGN KEY (Pers_Cedula)  
    REFERENCES SAPW_Persona(Pers_Cedula);
```

```
ALTER TABLE SAPW_Usuario ADD CONSTRAINT Usu_GrUs_Codigo_Fk  
    FOREIGN KEY (GrUs_Codigo)  
    REFERENCES SAPW_Grupo_Usuario(GrUS_Codigo);
```

```
ALTER TABLE SAPW_Rol_Web ADD CONSTRAINT RoWe_GrUs_Codigo_Fk  
    FOREIGN KEY (GrUs_Codigo)  
    REFERENCES SAPW_Grupo_Usuario(GrUs_Codigo);
```

```
ALTER TABLE SAPW_Rol_Sistema ADD CONSTRAINT RoSi_GrUs_Codigo_Fk  
    FOREIGN KEY (GrUs_Codigo)  
    REFERENCES SAPW_Grupo_Usuario(GrUs_Codigo);
```

```
ALTER TABLE SAPW_Perfil_Usuario ADD CONSTRAINT  
PeUs_OpSi_OpcionCodigo_Fk  
    FOREIGN KEY (OpSi_Codigo, OpSi_OpcionCodigo)  
    REFERENCES SAPW_Opcion_Sistema(OpSi_Codigo, OpSi_OpcionCodigo);
```

```
ALTER TABLE SAPW_Perfil_Usuario ADD CONSTRAINT PeUs_Esta_Codigo_Fk  
    FOREIGN KEY (Esta_Codigo)  
    REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Perfil_Usuario ADD CONSTRAINT PeUs_Usua_Login_Fk  
    FOREIGN KEY (Usua_Login)  
    REFERENCES SAPW_Usuario(Usua_Login);
```

```
ALTER TABLE SAPW_Perfil_Web ADD CONSTRAINT  
PeWe_OpWe_OpcionCodigo_Fk  
    FOREIGN KEY (OpWe_Codigo)  
    REFERENCES SAPW_Opcion_Web(OpWe_Codigo);
```

```
ALTER TABLE SAPW_Contenido ADD CONSTRAINT  
Cont_Usua_Login_Codigo_Fk  
    FOREIGN KEY (Usua_Login)  
    REFERENCES SAPW_Usuario(Usua_Login);
```

```
ALTER TABLE SAPW_Usuario_Forum ADD CONSTRAINT UsFo_Usua_Login_Fk  
    FOREIGN KEY (Usua_Login)  
    REFERENCES SAPW_Usuario(Usua_Login);
```

```
ALTER TABLE SAPW_Usuario_Forum ADD CONSTRAINT  
UsFo_Foru_Numero_Fk  
    FOREIGN KEY (Foru_Numero)  
    REFERENCES SAPW_Forum(Foru_Numero);
```

```
ALTER TABLE SAPW_Respuesta ADD CONSTRAINT Resp_Tema_Numero_Fk  
    FOREIGN KEY (Tema_Numero, Clas_Numero, Foru_Numero)  
    REFERENCES SAPW_TemaForum(Tema_Numero, Clas_Numero,  
Foru_Numero);
```

```
ALTER TABLE SAPW_Respuesta ADD CONSTRAINT Resp_Esta_Codigo_Fk  
    FOREIGN KEY (Esta_Codigo)  
    REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Clasificacion ADD CONSTRAINT Clas_Foru_Numero_Fk  
    FOREIGN KEY (Foru_Numero)  
    REFERENCES SAPW_Forum(Foru_Numero);
```

```
ALTER TABLE SAPW_TemaForum ADD CONSTRAINT Tema_Clas_Numero_Fk  
    FOREIGN KEY (Clas_Numero, Foru_Numero)  
    REFERENCES SAPW_Clasificacion(Clas_Numero, Foru_Numero);
```

```
ALTER TABLE SAPW_TemaForum ADD CONSTRAINT Tema_Esta_Codigo_Fk  
    FOREIGN KEY (Esta_Codigo)  
    REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Perfil_Empresarial ADD CONSTRAINT  
PeEm_Carr_Codgio_Fk  
    FOREIGN KEY (Carr_Codigo)  
    REFERENCES SAPW_Carrera(Carr_Codigo);
```



```
ALTER TABLE SAPW_Respuesta ADD CONSTRAINT ReFo_Esta_Codigo_Fk  
    FOREIGN KEY (Esta_Codigo)  
    REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Codigo_Fuente ADD CONSTRAINT CoFu_Usua_Login_Fk  
    FOREIGN KEY (Usua_Login)  
    REFERENCES SAPW_Usuario(Usua_Login);
```

```
ALTER TABLE SAPW_Codigo_Fuente ADD CONSTRAINT CoFu_Esta_Codigo_Fk  
    FOREIGN KEY (Esta_Codigo)  
    REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Cabecera_Auditoria ADD CONSTRAINT  
CaAu_Usua_Login_Fk  
    FOREIGN KEY (Usua_Login)  
    REFERENCES SAPW_Usuario(Usua_Login);
```

```
ALTER TABLE SAPW_Cabecera_Auditoria ADD CONSTRAINT  
CaAu_Esta_Codigo_Fk  
    FOREIGN KEY (Esta_Codigo)  
    REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Detalle_Auditoria ADD CONSTRAINT  
DeAu_CaAu_Codigo_Fk  
    FOREIGN KEY (CaAu_Codigo)  
    REFERENCES SAPW_Cabecera_Auditoria(CaAu_Codigo);
```

```
ALTER TABLE SAPW_Solicitud_Pasantia ADD CONSTRAINT  
SoPa_Usua_Login_Fk  
    FOREIGN KEY (Usua_Login)  
    REFERENCES SAPW_Usuario(Usua_Login);
```

```
ALTER TABLE SAPW_Usuario ADD CONSTRAINT Usua_Esta_Codigo_Fk  
    FOREIGN KEY (Esta_Codigo)  
    REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Solicitud_Pasantia ADD CONSTRAINT  
SoPa_Esta_Codigo_Fk  
    FOREIGN KEY (Esta_Codigo)  
    REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Pasantia ADD CONSTRAINT Pasa_Esta_Codigo_Fk  
    FOREIGN KEY (Esta_Codigo)  
    REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Reservacion_Laboratorio ADD CONSTRAINT  
ReLa_Labo_Codigo_Fk  
    FOREIGN KEY (Labo_Codigo)  
    REFERENCES SAPW_Laboratorio(Labo_Codigo);
```

```
ALTER TABLE SAPW_Reservacion_Laboratorio ADD CONSTRAINT  
ReLa_Usua_Login_Fk
```

```
    FOREIGN KEY (Usua_Login)  
    REFERENCES SAPW_Usuario(Usua_Login);
```

```
ALTER TABLE SAPW_Reservacion_Laboratorio ADD CONSTRAINT  
ReLa_Esta_Codigo_Fk
```

```
    FOREIGN KEY (Esta_Codigo)  
    REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Grupo_Usuario ADD CONSTRAINT GrUs_Esta_Codigo_Fk  
    FOREIGN KEY (Esta_Codigo)
```

```
    REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Rol_Sistema ADD CONSTRAINT DeRe_Esta_Codigo_Fk  
    FOREIGN KEY (Esta_Codigo)
```

```
    REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Rol_Sistema ADD CONSTRAINT RoSi_OpSi_Codigo_Fk  
    FOREIGN KEY (OpSi_Codigo, OpSi_OpcionCodigo)
```

```
    REFERENCES SAPW_Opcion_Sistema(OpSi_Codigo, OpSi_OpcionCodigo);
```

```
ALTER TABLE SAPW_Producto ADD CONSTRAINT Prod_Clie_Codigo_Fk  
    FOREIGN KEY (Clie_Codigo)
```

```
    REFERENCES SAPW_Cliente(Clie_Codigo);
```

```
ALTER TABLE SAPW_Producto ADD CONSTRAINT Prod_Serv_Numero_Fk  
    FOREIGN KEY (Serv_Numero, Depa_Codigo)
```

```
    REFERENCES SAPW_Servicio(Serv_Numero, Depa_Codigo);
```

```
ALTER TABLE SAPW_Servicio ADD CONSTRAINT Serv_Depa_Codigo_Fk  
    FOREIGN KEY (Depa_Codigo)
```

```
    REFERENCES SAPW_Departamento(Depa_Codigo);
```

```
ALTER TABLE SAPW_Tip ADD CONSTRAINT Tip_OpWe_Codigo_Fk  
    FOREIGN KEY (OpWe_Codigo)
```

```
    REFERENCES SAPW_Opcion_Web(OpWe_Codigo);
```

```
ALTER TABLE SAPW_Rol_Web ADD CONSTRAINT RoWe_OpWe_Codigo_Fk  
    FOREIGN KEY (OpWe_Codigo)
```

```
    REFERENCES SAPW_Opcion_Web(OpWe_Codigo);
```

```
ALTER TABLE SAPW_Rol_Web ADD CONSTRAINT RoWe_Esta_Codigo_Fk  
    FOREIGN KEY (Esta_Codigo)
```

```
    REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Empresa ADD CONSTRAINT Empe_Pers_Cedula_Fk  
    FOREIGN KEY (Pers_Cedula)
```

```
    REFERENCES SAPW_Persona(Pers_Cedula);
```



```
ALTER TABLE SAPW_Estudiante ADD CONSTRAINT Estu_Esta_Codigo_Fk  
    FOREIGN KEY (Esta_Codigo)  
    REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Curriculum_Vitae ADD CONSTRAINT  
CuVi_Pers_Cedula_Fk  
    FOREIGN KEY (Pers_Cedula)  
    REFERENCES SAPW_Persona(Pers_Cedula);
```

```
ALTER TABLE SAPW_Estudiante_Paralelo ADD CONSTRAINT  
EsPa_Estu_Matricula_Fk  
    FOREIGN KEY (Estu_Matricula)  
    REFERENCES SAPW_Estudiante(Estu_Matricula);
```

```
ALTER TABLE SAPW_Estudiante_Paralelo ADD CONSTRAINT  
EsPa_Esta_Codigo_Fk  
    FOREIGN KEY (Esta_Codigo)  
    REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Estudiante_Paralelo ADD CONSTRAINT  
EsPa_Para_Codigo_Fk  
    FOREIGN KEY (Anio_Codigo, Peri_Codigo, Para_Paralelo, NiAc_Codigo,  
Carr_Codigo, Mate_Codigo)  
    REFERENCES SAPW_Paralelo(Anio_Codigo, Peri_Codigo, Para_Paralelo,  
NiAc_Codigo, Carr_Codigo, Mate_Codigo);
```

```
ALTER TABLE SAPW_Respuesta_Estudiante ADD CONSTRAINT EsFo_Codigo_Fk  
    FOREIGN KEY (Estu_Matricula, NiAc_Codigo, Carr_Codigo, Mate_Codigo,  
Anio_Codigo, Peri_Codigo, Para_Paralelo)  
    REFERENCES SAPW_Estudiante_Paralelo(Estu_Matricula, NiAc_Codigo,  
Carr_Codigo, Mate_Codigo, Anio_Codigo, Peri_Codigo, Para_Paralelo);
```

```
ALTER TABLE SAPW_Nota ADD CONSTRAINT Nota_Codigo_Fk  
    FOREIGN KEY (Estu_Matricula, NiAc_Codigo, Carr_Codigo, Mate_Codigo,  
Anio_Codigo, Peri_Codigo, Para_Paralelo)  
    REFERENCES SAPW_Estudiante_Paralelo(Estu_Matricula, NiAc_Codigo,  
Carr_Codigo, Mate_Codigo, Anio_Codigo, Peri_Codigo, Para_Paralelo);
```

```
ALTER TABLE SAPW_Nota ADD CONSTRAINT Nota_Tipo_Codigo_Fk  
    FOREIGN KEY (Tipo_Codigo)  
    REFERENCES SAPW_Tipo(Tipo_Codigo);
```

```
ALTER TABLE SAPW_Desgloce ADD CONSTRAINT Desg_Codigo_Fk  
    FOREIGN KEY (Nota_Secuencial, Estu_Matricula, NiAc_Codigo,  
Carr_Codigo, Mate_Codigo, Anio_Codigo, Peri_Codigo, Para_Paralelo)  
    REFERENCES SAPW_Nota(Nota_Secuencial, Estu_Matricula, NiAc_Codigo,  
Carr_Codigo, Mate_Codigo, Anio_Codigo, Peri_Codigo, Para_Paralelo);
```

```
ALTER TABLE SAPW_Desgloce ADD CONSTRAINT Desg_Tipo_Codigo_Fk  
    FOREIGN KEY (Tipo_Codigo)  
    REFERENCES SAPW_Tipo(Tipo_Codigo);
```

```
ALTER TABLE SAPW_Desgloce_Trabajo ADD CONSTRAINT DeTr_Codigo_Fk  
    FOREIGN KEY (Desg_Linea, Nota_Secuencial, Estu_Matricula, NiAc_Codigo,  
    Carr_Codigo, Mate_Codigo, Anio_Codigo, Peri_Codigo, Para_Paralelo)  
    REFERENCES SAPW_Desgloce(Desg_Linea, Nota_Secuencial,  
    Estu_Matricula, NiAc_Codigo, Carr_Codigo, Mate_Codigo, Anio_Codigo,  
    Peri_Codigo, Para_Paralelo);
```

```
ALTER TABLE SAPW_Docente_Carrera ADD CONSTRAINT  
DoCa_Carr_Codigo_Fk  
    FOREIGN KEY (Carr_Codigo)  
    REFERENCES SAPW_Carrera(Carr_Codigo);
```

```
ALTER TABLE SAPW_Nivel_Academico ADD CONSTRAINT  
NiAc_Carr_Codigo_Fk  
    FOREIGN KEY (Carr_Codigo)  
    REFERENCES SAPW_Carrera(Carr_Codigo);
```

```
ALTER TABLE SAPW_Materia_Nivel ADD CONSTRAINT MaNi_NiAc_Codigo_Fk  
    FOREIGN KEY (NiAc_Codigo, Carr_Codigo)  
    REFERENCES SAPW_Nivel_Academico(NiAc_Codigo, Carr_Codigo);
```

```
ALTER TABLE SAPW_Materia_Nivel ADD CONSTRAINT MaNi_Mate_Codigo_Fk  
    FOREIGN KEY (Mate_Codigo)  
    REFERENCES SAPW_Materia(Mate_Codigo);
```

```
ALTER TABLE SAPW_Paralelo ADD CONSTRAINT Para_MaNi_Codigo_Fk  
    FOREIGN KEY (NiAc_Codigo, Carr_Codigo, Mate_Codigo)  
    REFERENCES SAPW_Materia_Nivel(NiAc_Codigo, Carr_Codigo,  
    Mate_Codigo);
```

```
ALTER TABLE SAPW_Paralelo ADD CONSTRAINT Para_Peri_Codigo_Fk  
    FOREIGN KEY (Peri_Codigo, Carr_Codigo, Anio_Codigo)  
    REFERENCES SAPW_Periodo_Anio_Carrera(Peri_Codigo, Carr_Codigo,  
    Anio_Codigo);
```

```
ALTER TABLE SAPW_Politica ADD CONSTRAINT Poli_Para_Codigo_Fk  
    FOREIGN KEY (Anio_Codigo, Peri_Codigo, Para_Paralelo, NiAc_Codigo,  
    Carr_Codigo, Mate_Codigo)  
    REFERENCES SAPW_Paralelo(Anio_Codigo, Peri_Codigo, Para_Paralelo,  
    NiAc_Codigo, Carr_Codigo, Mate_Codigo);
```

```
ALTER TABLE SAPW_Capitulo_Estudio ADD CONSTRAINT  
CaEs_Para_Codigo_Fk  
    FOREIGN KEY (Anio_Codigo, Peri_Codigo, Para_Paralelo, NiAc_Codigo,  
    Carr_Codigo, Mate_Codigo)
```

```
REFERENCES SAPW_Paralelo(Anio_Codigo, Peri_Codigo, Para_Paralelo,  
NiAc_Codigo, Carr_Codigo, Mate_Codigo);
```

```
ALTER TABLE SAPW_Deber ADD CONSTRAINT Deb_CaEs_Codigo_Fk  
FOREIGN KEY (CaEs_Numero, Anio_Codigo, Peri_Codigo, Para_Paralelo,  
NiAc_Codigo, Carr_Codigo, Mate_Codigo)  
REFERENCES SAPW_Capitulo_Estudio(CaEs_Numero, Anio_Codigo,  
Peri_Codigo, Para_Paralelo, NiAc_Codigo, Carr_Codigo, Mate_Codigo);
```

```
ALTER TABLE SAPW_Pasantia ADD CONSTRAINT Pasa_SoPa_Codigo_Fk  
FOREIGN KEY (SoPa_Codigo)  
REFERENCES SAPW_Solicitud_Pasantia(SoPa_Codigo);
```

```
ALTER TABLE SAPW_Seminario ADD CONSTRAINT Semi_Cert_Codigo_Fk  
FOREIGN KEY (Cert_Codigo)  
REFERENCES SAPW_Certificado(Cert_Codigo);
```

```
ALTER TABLE SAPW_Seminario ADD CONSTRAINT Semi_Esta_Codigo_Fk  
FOREIGN KEY (Esta_Codigo)  
REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Horario_Seminario ADD CONSTRAINT  
HoSe_Hora_Codigo_Fk  
FOREIGN KEY (Hora_Codigo)  
REFERENCES SAPW_Horario(Hora_Codigo);
```

```
ALTER TABLE SAPW_Horario_Paralelo ADD CONSTRAINT  
HoPa_Hora_Codigo_Fk  
FOREIGN KEY (Hora_Codigo)  
REFERENCES SAPW_Horario(Hora_Codigo);
```

```
ALTER TABLE SAPW_Horario_Paralelo ADD CONSTRAINT  
HoSe_Para_Codigo_Fk  
FOREIGN KEY (Anio_Codigo, Peri_Codigo, Para_Paralelo, NiAc_Codigo,  
Carr_Codigo, Mate_Codigo)  
REFERENCES SAPW_Paralelo(Anio_Codigo, Peri_Codigo, Para_Paralelo,  
NiAc_Codigo, Carr_Codigo, Mate_Codigo);
```

```
ALTER TABLE SAPW_Sonido ADD CONSTRAINT Soni_EsAn_Codigo_Fk  
FOREIGN KEY (EsAn_Codigo)  
REFERENCES SAPW_Estado_Animo (EsAn_Codigo);
```

```
ALTER TABLE SAPW_Colores_Animo ADD CONSTRAINT  
CoAn_EsAn_Codigo_Fk  
FOREIGN KEY (EsAn_Codigo)  
REFERENCES SAPW_Estado_Animo (EsAn_Codigo);
```

```
ALTER TABLE SAPW_Part es_Pagina ADD CONSTRAINT PaPa_EsAn_Codigo_Fk  
FOREIGN KEY (CoAn_Codigo, EsAn_Codigo)  
REFERENCES SAPW_Colores_Animo (CoAn_Codigo, EsAn_Codigo);
```

```
ALTER TABLE SAPW_Tema ADD CONSTRAINT Tema_Regl_Codigo_Fk
    FOREIGN KEY (Regl_Codigo)
    REFERENCES SAPW_Reglamento (Regl_Codigo);

ALTER TABLE SAPW_Articulo ADD CONSTRAINT Arti_Regl_Codigo_Fk
    FOREIGN KEY (Tema_Codigo, Regl_Codigo)
    REFERENCES SAPW_Tema (Tema_Codigo, Regl_Codigo);

ALTER TABLE SAPW_Anuncio ADD CONSTRAINT Anun_Esta_Codigo_Fk
    FOREIGN KEY (Esta_Codigo)
    REFERENCES SAPW_Estado (Esta_Codigo);

ALTER TABLE SAPW_Directivo ADD CONSTRAINT Dire_Divi_Codigo_Fk
    FOREIGN KEY (Divi_Codigo)
    REFERENCES SAPW_Division (Divi_Codigo);

ALTER TABLE SAPW_Directivo ADD CONSTRAINT Dire_Carg_Codigo_Fk
    FOREIGN KEY (Carg_Codigo)
    REFERENCES SAPW_Cargo (Carg_Codigo);

ALTER TABLE SAPW_Delegado ADD CONSTRAINT Dele_Codigo_Fk
    FOREIGN KEY (Dire_Numero, Divi_Codigo)
    REFERENCES SAPW_Directivo (Dire_Numero, Divi_Codigo);

ALTER TABLE SAPW_Delegado ADD CONSTRAINT Dele_Carg_Codigo_Fk
    FOREIGN KEY (Carg_Codigo)
    REFERENCES SAPW_Cargo (Carg_Codigo);

ALTER TABLE SAPW_Integrante_Resolucion ADD CONSTRAINT
InRe_CaRe_Codigo_Fk
    FOREIGN KEY (CaRe_Codigo)
    REFERENCES SAPW_Cabecera_Resolucion (CaRe_Codigo);

ALTER TABLE SAPW_Detalle_Resolucion ADD CONSTRAINT
DeRe_CaRe_Codigo_Fk
    FOREIGN KEY (CaRe_Codigo)
    REFERENCES SAPW_Cabecera_Resolucion (CaRe_Codigo);

ALTER TABLE SAPW_Horario_Seminario ADD CONSTRAINT
HoSe_Cert_Codigo_Fk
    FOREIGN KEY (Semi_Codigo, Cert_Codigo)
    REFERENCES SAPW_Seminario (Semi_Codigo, Cert_Codigo);

ALTER TABLE SAPW_Deuda_Estudiente ADD CONSTRAINT
DeEs_CodigoEstudiante_Fk
    FOREIGN KEY (EsTu_Matricula)
    REFERENCES SAPW_Estudiente (Estu_Matricula);

ALTER TABLE SAPW_Deuda_Estudiente ADD CONSTRAINT
DeEs_CodigoCarrera_Fk
```

```
FOREIGN KEY (Carr_Codigo)
REFERENCES SAPW_Carrera (Carr_Codigo);
```

```
ALTER TABLE SAPW_Anio_Carrera ADD CONSTRAINT Anio_Carr_Codigo_Fk
FOREIGN KEY (Carr_Codigo)
REFERENCES SAPW_Carrera(Carr_Codigo);
```

```
ALTER TABLE SAPW_Anio_Carrera ADD CONSTRAINT Anio_Anio_Codigo_Fk
FOREIGN KEY (Anio_Codigo)
REFERENCES SAPW_Anio(Anio_Codigo);
```

```
ALTER TABLE SAPW_Anio_Carrera ADD CONSTRAINT Anio_Esta_Codigo_Fk
FOREIGN KEY (Esta_Codigo)
REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Anio_Carrera ADD CONSTRAINT Anio_EstadoAnio_Fk
FOREIGN KEY (Esta_Anio)
REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Periodo_Anio_Carrera ADD CONSTRAINT
Peri_Anio_Codigo_Fk
FOREIGN KEY (Anio_Codigo,Carr_Codigo)
REFERENCES SAPW_Anio_Carrera(Anio_Codigo,Carr_Codigo);
```

```
ALTER TABLE SAPW_Estudiante_Carrera ADD CONSTRAINT
EsCa_Estu_Matricula_Fk
FOREIGN KEY (Estu_Matricula)
REFERENCES SAPW_Estudiante(Estu_Matricula);
```

```
ALTER TABLE SAPW_Estudiante_Carrera ADD CONSTRAINT
EsCa_Carr_Codigo_Fk
FOREIGN KEY (Carr_Codigo)
REFERENCES SAPW_Carrera(Carr_Codigo);
```

```
ALTER TABLE SAPW_Estudiante_Carrera ADD CONSTRAINT
EsCa_Esta_Codigo_Fk
FOREIGN KEY (Esta_Codigo)
REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Promocion ADD CONSTRAINT Prom_Anio_Codigo_Fk
FOREIGN KEY (Anio_Codigo, Carr_Codigo)
REFERENCES SAPW_Anio_Carrera (Anio_Codigo, Carr_Codigo);
```

```
ALTER TABLE SAPW_Periodo_Anio_Carrera ADD CONSTRAINT Peri_Estado_Fk
FOREIGN KEY (Esta_Codigo)
REFERENCES SAPW_Estado (Esta_Codigo);
```

```
ALTER TABLE SAPW_Periodo_Anio_Carrera ADD CONSTRAINT
Peri_EstadoPeriodo_Fk
```



```
FOREIGN KEY (Esta_Periodo)
REFERENCES SAPW_Estado (Esta_Codigo);
```

```
ALTER TABLE SAPW_Historia_Estudiante ADD CONSTRAINT
HiEs_Esta_Codigo_Fk
FOREIGN KEY (Esta_Codigo)
REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Capacitacion_In_Situ ADD CONSTRAINT
CaIs_Esta_Codigo_Fk
FOREIGN KEY (Esta_Codigo)
REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Diccionario ADD CONSTRAINT Dicc_Esta_Codigo_Fk
FOREIGN KEY (Esta_Codigo)
REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Opcion_Web ADD CONSTRAINT OpWe_Esta_Codigo_Fk
FOREIGN KEY (Esta_Codigo)
REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Pasantia ADD CONSTRAINT SoPa_Depa_Codigo_Fk
FOREIGN KEY (Depa_Codigo)
REFERENCES SAPW_Departamento(Depa_Codigo);
```

```
ALTER TABLE SAPW_Solicitud_Pasantia ADD CONSTRAINT
SoPa_Depar_Codigo_Fk
FOREIGN KEY (Depa_Codigo)
REFERENCES SAPW_Departamento(Depa_Codigo);
```

```
ALTER TABLE SAPW_Buzon ADD CONSTRAINT BuSu_Tipo_Codigo_Fk
FOREIGN KEY (BuSu_Tipo)
REFERENCES SAPW_Tipo(Tipo_Codigo);
```

```
ALTER TABLE SAPW_Persona ADD CONSTRAINT Pers_Esta_Codigo_Fk
FOREIGN KEY (Esta_Codigo)
REFERENCES SAPW_Estado(Esta_Codigo);
```

```
ALTER TABLE SAPW_Usuario ADD CONSTRAINT Usua_Pers_Cedula_Fk
FOREIGN KEY (Usua_Propietario)
REFERENCES SAPW_Persona(Pers_Cedula);
```

```
ALTER TABLE SAPW_Parametro ADD CONSTRAINT Para_Pers_Cedula_Fk
FOREIGN KEY (Pers_Cedula)
REFERENCES SAPW_Persona(Pers_Cedula);
```

```
ALTER TABLE SAPW_Evento ADD CONSTRAINT Even_Esta_Codigo_Fk
FOREIGN KEY (Esta_Codigo)
REFERENCES SAPW_Estado (Esta_Codigo);
```

```
ALTER TABLE SAPW_Preguntas_Testimonios ADD CONSTRAINT  
Esta_Preguntas_Fk  
    FOREIGN KEY (Esta_Codigo) REFERENCES SAPW_Estado (Esta_Codigo);
```

```
ALTER TABLE SAPW_Testimonios ADD CONSTRAINT Esta_Testimonios_Fk  
    FOREIGN KEY (Esta_Codigo)  
    REFERENCES SAPW_Estado (Esta_Codigo);
```

```
ALTER TABLE SAPW_Detalle_Testimonios ADD CONSTRAINT  
Esta_Detalle_Test_Fk  
    FOREIGN KEY (Esta_Codigo)  
    REFERENCES SAPW_Estado (Esta_Codigo);
```

```
ALTER TABLE SAPW_Detalle_Testimonios ADD CONSTRAINT Preg_Numero_Fk  
    FOREIGN KEY (Preg_Numero)  
    REFERENCES SAPW_Preguntas_Testimonios(Preg_Numero);
```

```
ALTER TABLE SAPW_Detalle_Testimonios ADD CONSTRAINT Test_Numero_Fk  
    FOREIGN KEY (Test_Numero)  
    REFERENCES SAPW_Testimonios(Test_Numero);
```

```
ALTER TABLE SAPW_Detalle_Fondo_Beca ADD CONSTRAINT  
DeBe_Esta_Codigo_Fk  
    FOREIGN KEY (Esta_Codigo)  
    REFERENCES SAPW_Estado (Esta_Codigo);
```

```
ALTER TABLE SAPW_Detalle_Fondo_Beca ADD CONSTRAINT  
DeBe_fobe_Codigo_Fk  
    FOREIGN KEY (Fobe_Codigo)  
    REFERENCES SAPW_Fondo_beca (fobe_Codigo);
```

```
ALTER TABLE SAPW_Requisito_Beca ADD CONSTRAINT rebe_fobe_codigo_fk  
    FOREIGN KEY (Fobe_Codigo)  
    REFERENCES SAPW_Fondo_beca (fobe_Codigo);
```

```
ALTER TABLE SAPW_Requisito_Pasantia ADD CONSTRAINT  
repa_carr_codigo_fk  
    FOREIGN KEY (Carr_Codigo)  
    REFERENCES SAPW_Carrera (Carr_Codigo);
```

```
ALTER TABLE SAPW_Requisito_Pasantia ADD CONSTRAINT  
repa_esta_codigo_fk  
    FOREIGN KEY (Esta_Codigo)  
    REFERENCES SAPW_Estado (Esta_Codigo);
```

```
ALTER TABLE SAPW_Materia_Req_Pasantia ADD CONSTRAINT  
mapa_mate_codigo_fk  
    FOREIGN KEY (mapa_Codigo)  
    REFERENCES SAPW_materia (Mate_Codigo);
```



```
ALTER TABLE SAPW_Politica_Pasantia ADD CONSTRAINT popa_carr_codigo_fk  
    FOREIGN KEY (carr_Codigo)  
    REFERENCES SAPW_Carrera (carr_Codigo);
```

```
ALTER TABLE SAPW_Politica_Pasantia ADD CONSTRAINT popa_esta_codigo_fk  
    FOREIGN KEY (esta_Codigo)  
    REFERENCES SAPW_Estado (esta_Codigo);
```

```
ALTER TABLE SAPW_Pasantias_Empresa ADD CONSTRAINT  
pasem_carr_codigo_fk  
    FOREIGN KEY (carr_Codigo)  
    REFERENCES SAPW_Carrera (carr_Codigo);
```

```
ALTER TABLE SAPW_Pasantias_Empresa ADD CONSTRAINT  
pasem_depa_codigo_fk  
    FOREIGN KEY (depa_Codigo)  
    REFERENCES SAPW_Departamento (depa_Codigo);
```

```
ALTER TABLE SAPW_Pasantias_Empresa ADD CONSTRAINT  
pasem_empr_codigo_fk  
    FOREIGN KEY (empr_ruc)  
    REFERENCES SAPW_Empresa (emp_ruc);
```

4.1.4 Creación de Índices

```
CREATE INDEX Dicc_FraseClave  
    ON SAPW_Diccionario (Dicc_Frase);
```

4.1.5 Creación de Secuencias

```
DROP SEQUENCE SAPW_S_Anuncios;  
CREATE SEQUENCE SAPW_S_Anuncios START 1;
```

```
DROP SEQUENCE SAPW_S_Carreras;  
CREATE SEQUENCE SAPW_S_Carreras START 1;
```

```
DROP SEQUENCE SAPW_S_Certificados;  
CREATE SEQUENCE SAPW_S_Certificados START 1;
```

```
DROP SEQUENCE SAPW_S_Seminarios;  
CREATE SEQUENCE SAPW_S_Seminarios START 1;
```

```
DROP SEQUENCE SAPW_S_Cabecera_Auditoria;  
CREATE SEQUENCE SAPW_S_Cabecera_Auditoria START 1;
```

DROP SEQUENCE SAPW_S_Detalle_Auditoria;
CREATE SEQUENCE SAPW_S_Detalle_Auditoria START 1;

DROP SEQUENCE SAPW_S_Clientes;
CREATE SEQUENCE SAPW_S_Clientes START 1;

DROP SEQUENCE SAPW_S_Titulo_Profesional;
CREATE SEQUENCE SAPW_S_Titulo_Profesional START 1;

DROP SEQUENCE SAPW_S_Link;
CREATE SEQUENCE SAPW_S_Link START 1;

DROP SEQUENCE SAPW_S_Solicitud_Personal;
CREATE SEQUENCE SAPW_S_Solicitud_Personal START 1;

DROP SEQUENCE SAPW_S_Catalogo_Protcom;
CREATE SEQUENCE SAPW_S_Catalogo_Protcom START 1;

DROP SEQUENCE SAPW_S_Departamento;
CREATE SEQUENCE SAPW_S_Departamento START 1;

DROP SEQUENCE SAPW_S_Modulos;
CREATE SEQUENCE SAPW_S_Modulos START 1;

DROP SEQUENCE SAPW_S_Servicios;
CREATE SEQUENCE SAPW_S_Servicios START 1;

DROP SEQUENCE SAPW_S_Instituciones;
CREATE SEQUENCE SAPW_S_Instituciones START 1;

DROP SEQUENCE SAPW_S_Productos;
CREATE SEQUENCE SAPW_S_Productos START 1;

DROP SEQUENCE SAPW_S_Opcion_Web;
CREATE SEQUENCE SAPW_S_Opcion_Web START 1;

DROP SEQUENCE SAPW_S_Tips;
CREATE SEQUENCE SAPW_S_Tips START 1;

DROP SEQUENCE SAPW_S_Cargos;
CREATE SEQUENCE SAPW_S_Cargos START 1;

DROP SEQUENCE SAPW_S_Recurso_Humano;
CREATE SEQUENCE SAPW_S_Recurso_Humano START 1;

DROP SEQUENCE SAPW_S_Nivel_Academico;
CREATE SEQUENCE SAPW_S_Nivel_Academico START 1;

DROP SEQUENCE SAPW_S_Diccionarios;
CREATE SEQUENCE SAPW_S_Diccionarios START 1;

DROP SEQUENCE SAPW_S_Grupo_Libros;
CREATE SEQUENCE SAPW_S_Grupo_Libros START 1;

DROP SEQUENCE SAPW_S_Libros;
CREATE SEQUENCE SAPW_S_Libros START 1;
DROP SEQUENCE SAPW_S_Políticas;
CREATE SEQUENCE SAPW_S_Políticas START 1;

DROP SEQUENCE SAPW_S_Capitulos_Estudio;
CREATE SEQUENCE SAPW_S_Capitulos_Estudio START 1;

DROP SEQUENCE SAPW_S_Reglamentos;
CREATE SEQUENCE SAPW_S_Reglamentos START 1;

DROP SEQUENCE SAPW_S_Temas;
CREATE SEQUENCE SAPW_S_Temas START 1;

DROP SEQUENCE SAPW_S_Articulos;
CREATE SEQUENCE SAPW_S_Articulos START 1;

DROP SEQUENCE SAPW_S_PolíticasReservacion;
CREATE SEQUENCE SAPW_S_PolíticasReservacion START 1;

DROP SEQUENCE SAPW_S_Salones_Clase;
CREATE SEQUENCE SAPW_S_Salones_clase START 1;

DROP SEQUENCE SAPW_S_Laboratorio;
CREATE SEQUENCE SAPW_S_Laboratorio START 1;

DROP SEQUENCE SAPW_S_Horario;
CREATE SEQUENCE SAPW_S_Horario START 1;

DROP SEQUENCE SAPW_S_Cabecera_Resolucion;
CREATE SEQUENCE SAPW_S_Cabecera_Resolucion START 1;

DROP SEQUENCE SAPW_S_Detalle_Resolucion;
CREATE SEQUENCE SAPW_S_Detalle_Resolucion START 1;

DROP SEQUENCE SAPW_S_Grupo_Usuario;
CREATE SEQUENCE SAPW_S_Grupo_Usuario START 1;

DROP SEQUENCE SAPW_S_Cabecera_Guia_Telefono;
CREATE SEQUENCE SAPW_S_Cabecera_Guia_Telefono START 1;

DROP SEQUENCE SAPW_S_Detalle_Guia_Telefono;
CREATE SEQUENCE SAPW_S_Detalle_Guia_Telefono START 1;

DROP SEQUENCE SAPW_S_Evento;
CREATE SEQUENCE SAPW_S_Evento START 1;

DROP SEQUENCE SAPW_S_Paralelos;
CREATE SEQUENCE SAPW_S_Paralelos START 1;

DROP SEQUENCE SAPW_S_Perfil_Empresarial;
CREATE SEQUENCE SAPW_S_Perfil_Empresarial START 1;
DROP SEQUENCE SAPW_S_Division;
CREATE SEQUENCE SAPW_S_Division START 1;

DROP SEQUENCE SAPW_S_Directivo;
CREATE SEQUENCE SAPW_S_Directivo START 1;

DROP SEQUENCE SAPW_S_Delegado;
CREATE SEQUENCE SAPW_S_Delegado START 1;

DROP SEQUENCE SAPW_S_Codigo_Fuente;
CREATE SEQUENCE SAPW_S_Codigo_Fuente START 1;

DROP SEQUENCE SAPW_S_Horario_Seminario;
CREATE SEQUENCE SAPW_S_Horario_Seminario START 1;

DROP SEQUENCE SAPW_S_Forum;
CREATE SEQUENCE SAPW_S_Forum START 1;

DROP SEQUENCE SAPW_S_TemaForum;
CREATE SEQUENCE SAPW_S_TemaForum START 1;

DROP SEQUENCE SAPW_S_Respuesta;
CREATE SEQUENCE SAPW_S_Respuesta START 1;

DROP SEQUENCE SAPW_S_Contacto;
CREATE SEQUENCE SAPW_S_Contacto START 1;

DROP SEQUENCE SAPW_S_Pc;
CREATE SEQUENCE SAPW_S_Pc START 1;

DROP SEQUENCE SAPW_S_Generacion_Pc;
CREATE SEQUENCE SAPW_S_Generacion_Pc START 1;

DROP SEQUENCE SAPW_S_Anio;
CREATE SEQUENCE SAPW_S_Anio START 1;

DROP SEQUENCE SAPW_S_Periodo;
CREATE SEQUENCE SAPW_S_Periodo START 1;

DROP SEQUENCE SAPW_S_Forum_Clase;
CREATE SEQUENCE SAPW_S_Forum_Clase START 1;

DROP SEQUENCE SAPW_S_Pregunta_Estudiante;
CREATE SEQUENCE SAPW_S_Pregunta_Estudiante START 1;

```
DROP SEQUENCE SAPW_S_Promocion;  
CREATE SEQUENCE SAPW_S_Promocion START 1;
```

```
DROP SEQUENCE SAPW_S_Deber;  
CREATE SEQUENCE SAPW_S_Deber START 1;  
DROP SEQUENCE SAPW_S_Nota;  
CREATE SEQUENCE SAPW_S_Nota START 1;
```

```
DROP SEQUENCE SAPW_S_Desgloce;  
CREATE SEQUENCE SAPW_S_Desgloce START 1;
```

```
DROP SEQUENCE SAPW_S_Buzon;  
CREATE SEQUENCE SAPW_S_Buzon START 1;
```

```
DROP SEQUENCE SAPW_S_Desgloce_Trabajo;  
CREATE SEQUENCE SAPW_S_Desgloce_Trabajo START 1;
```

```
DROP SEQUENCE SAPW_S_Foto_Cabecera;  
CREATE SEQUENCE SAPW_S_Foto_Cabecera START 1;
```

```
DROP SEQUENCE SAPW_S_Foto_Detalle;  
CREATE SEQUENCE SAPW_S_Foto_Detalle START 1;
```

```
DROP SEQUENCE SAPW_S_Deuda_Estudiante;  
CREATE SEQUENCE SAPW_S_Deuda_Estudiante START 1;
```

```
DROP SEQUENCE SAPW_S_Documentos;  
CREATE SEQUENCE SAPW_S_Documentos START 1;
```

```
DROP SEQUENCE SAPW_S_Cabecera_Auditoria;  
CREATE SEQUENCE SAPW_S_Cabecera_Auditoria START 1;
```

```
DROP SEQUENCE SAPW_S_Detalle_Auditoria;  
CREATE SEQUENCE SAPW_S_Detalle_Auditoria START 1;
```

```
DROP SEQUENCE SAPW_S_Observaciones;  
CREATE SEQUENCE SAPW_S_Observaciones START 1;
```

```
DROP SEQUENCE SAPW_S_Proyecto;  
CREATE SEQUENCE SAPW_S_Proyecto START 1;
```

```
DROP SEQUENCE SAPW_S_Avance;  
CREATE SEQUENCE SAPW_S_Avance START 1;
```

```
DROP SEQUENCE SAPW_S_Capacitacion_In_Situ;  
CREATE SEQUENCE SAPW_S_Capacitacion_In_Situ START 1;
```

```
DROP SEQUENCE SAPW_S_Solicitud_Pasantia;  
CREATE SEQUENCE SAPW_S_Solicitud_Pasantia START 1;
```

```
DROP SEQUENCE SAPW_S_Reservacion_Laboratorios;
CREATE SEQUENCE SAPW_S_Reservacion_Laboratorios START 1;
```

```
DROP SEQUENCE SAPW_S_BackupOnLine;
CREATE SEQUENCE SAPW_S_BackupOnLine START 1;
```

```
DROP SEQUENCE SAPW_S_Clasificacion;
CREATE SEQUENCE SAPW_S_Clasificacion START 1;
```

```
DROP SEQUENCE SAPW_S_Beca;
CREATE SEQUENCE SAPW_S_Beca START 1;
```

```
DROP SEQUENCE SAPW_S_Detalle_Beca;
CREATE SEQUENCE SAPW_S_Detalle_Beca START 1;
```

```
DROP SEQUENCE SAPW_S_Preguntas_Testimonios;
CREATE SEQUENCE SAPW_S_Preguntas_Testimonios START 1;
```

```
DROP SEQUENCE SAPW_S_Requisito_Beca;
CREATE SEQUENCE SAPW_S_Requisito_Beca START 1;
```

```
DROP SEQUENCE SAPW_S_Testimonios;
CREATE SEQUENCE SAPW_S_Testimonios START 1;
```



4.1.6 Creación de Funciones

--Función para el Mantenimiento de Anuncios

```
DROP FUNCTION SAPW_Mant_Anuncios(int4, int4, int4, varchar, varchar, date,
varchar)
CREATE FUNCTION public.sapw_mant_anuncios(int4, int4, int4, varchar, varchar,
date, varchar) RETURNS int4 AS '
```

```
DECLARE
```

```
    V_Cab_Auditoria    ALIAS FOR $1;
    V_Anun_Codigo      ALIAS FOR $2;
    V_Esta_codigo      ALIAS FOR $3;
    V_Anun_Titulo      ALIAS FOR $4;
    V_Anun_Detalle     ALIAS FOR $5;
    V_Anun_Fecha       ALIAS FOR $6;
    V_Tipo              ALIAS FOR $7;
    V_Secuencia INTEGER;
```

```
BEGIN
```

```
    IF V_Tipo = "I" THEN
```

```
        SELECT NEXTVAL ("SAPW_S_Anuncios") INTO V_Secuencia;
        INSERT INTO SAPW_Anuncio VALUES(V_Secuencia, V_Esta_Codigo,
        V_Anun_Titulo, V_Anun_Detalle, V_Anun_Fecha, NEXTVAL
        ("SAPW_S_Foto_Cabecera"));
        INSERT INTO SAPW_Detalle_Auditoria VALUES
        (NEXTVAL("SAPW_S_Detalle_Auditoria"), V_Cab_Auditoria, "En el
```



```

Mantenimiento de Anuncios se Creo el registro con Código " ||
V_Secuencia::TEXT, "Se trabajo en la tabla de Anuncios
(SAPW_Anuncio)",(SELECT Current_TimeStamp));
ELSE
  V_Secuencia := V_Anun_Codigo;
  IF V_Tipo = "U" THEN
    UPDATE SAPW_Anuncio SET Esta_codigo = V_Esta_Codigo,
    Anun_Titulo = V_Anun_Titulo, Anun_Detalle =
    V_Anun_Detalle, Anun_FechaPublicacion = V_Anun_Fecha
    WHERE Anun_Codigo = V_Anun_Codigo;
    INSERT INTO SAPW_Detalle_Auditoria VALUES
    (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"
    En el Mantenimiento de Anuncios se Modificó el registro con
    Código " || V_Anun_Codigo::TEXT, "Se trabajo en la tabla de
    Anuncios (SAPW_Anuncio)",(SELECT Current_TimeStamp));
  ELSE
    IF V_Tipo = "D" THEN
      DELETE FROM SAPW_Anuncio WHERE Anun_Codigo
      = V_Anun_Codigo;
      INSERT INTO SAPW_Detalle_Auditoria VALUES
      (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Audi
      toria,"En el Mantenimiento de Anuncios se Eliminó el
      registro con Código " || V_Anun_Codigo::TEXT, "Se
      trabajo en la tabla de Anuncios
      (SAPW_Anuncio)",(SELECT Current_TimeStamp));
    END IF;
  END IF;
END IF;
...UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
...(SELECT Current_Date);
...RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Cargos

```

DROP FUNCTION SAPW_Mant_Cargos(INTEGER,INTEGER, VARCHAR,
VARCHAR);
CREATE OR REPLACE FUNCTION SAPW_Mant_Cargos(INTEGER,INTEGER,
VARCHAR, VARCHAR) RETURNS INTEGER AS '
  DECLARE
    V_Cab_Auditoria          ALIAS FOR $1;
    V_Carg_Codigo            ALIAS FOR $2;
    V_Carg_Descripcion        ALIAS FOR $3;
    V_Tipo                   ALIAS FOR $4;
    V_Secuencia              INTEGER;
  BEGIN
    IF V_Tipo = "I" THEN
      SELECT NEXTVAL ("SAPW_S_Cargos") INTO V_Secuencia;

```



```

INSERT INTO SAPW_Cargo VALUES(V_Secuencia,
V_Carg_Descripcion);
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Cargos se Creo el registro con Código " || V_Secuencia::TEXT,
"Se trabajo en la tabla de Cargos (SAPW_Cargo)",(SELECT Current_TimeStamp));
ELSE
V_Secuencia := V_Carg_Codigo;
IF V_Tipo = "U" THEN
UPDATE SAPW_Cargo SET Carg_Descripcion =
V_Carg_Descripcion WHERE Carg_Codigo = V_Carg_Codigo;
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Cargos se Modificó el registro con Código " || V_Carg_Codigo::TEXT,
"Se trabajo en la tabla de Cargos (SAPW_Cargo)",(SELECT Current_TimeStamp));
ELSE
IF V_Tipo = "D" THEN
DELETE FROM SAPW_Cargo WHERE Carg_Codigo = V_Carg_Codigo;
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Cargos se Eliminó el registro con Código " || V_Carg_Codigo::TEXT,
"Se trabajo en la tabla de Cargos (SAPW_Cargo)",(SELECT Current_TimeStamp));
END IF;
END IF;
END IF;
RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Titulos Profesionales

```

DROP FUNCTION SAPW_Mant_TitulosProfesionales(INTEGER,INTEGER,
VARCHAR, VARCHAR, VARCHAR, VARCHAR);
CREATE OR REPLACE FUNCTION
SAPW_Mant_TitulosProfesionales(INTEGER,INTEGER, VARCHAR, VARCHAR,
VARCHAR, VARCHAR) RETURNS INTEGER AS '

```

```

DECLARE

```

```

V_Cab_Auditoria          ALIAS FOR $1;
V_TiPr_Codigo            ALIAS FOR $2;
V_TiPr_Nombre            ALIAS FOR $3;
V_TiPr_Descripción       ALIAS FOR $4;
V_TiPr_Abreviatura       ALIAS FOR $5;
V_Tipo                   ALIAS FOR $6;
V_Secuencia              INTEGER;

```

```

BEGIN

```

```

IF V_Tipo = "I" THEN
SELECT NEXTVAL ("SAPW_S_Titulo_Profesional") INTO V_Secuencia;
INSERT INTO SAPW_Titulo_Profesional VALUES(V_Secuencia,
V_TiPr_Nombre, V_TiPr_Descripcion, V_TiPr_Abreviatura);

```

```

INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Titulos Profesionales se Creo el registro con Código " || V_Secuencia::TEXT,
"Se trabajo en la tabla de Titulos Profesionales (SAPW_Titulo_Profesional)",(SELECT
Current_TimeStamp));
ELSE
    V_Secuencia := V_TiPr_Codigo;
    IF V_Tipo = "U" THEN
UPDATE SAPW_Titulo_Profesional SET TiPr_Descripcion = V_TiPr_Descripcion,
TiPr_Nombre = V_TiPr_Nombre, TiPr_Abreviatura = V_TiPr_Abreviatura
WHERE TiPr_Codigo = V_TiPr_Codigo;
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Titulos Profesionales se Modificó el registro con Código " || V_TiPr_Codigo::TEXT,
"Se trabajo en la tabla de Titulos Profesionales (SAPW_Titulo_Profesional)",(SELECT
Current_TimeStamp));
ELSE
    IF V_Tipo = "D" THEN
DELETE FROM SAPW_Titulo_Profesional WHERE TiPr_Codigo = V_TiPr_Codigo;
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Titulos Profesionales se Eliminó el registro con Código " || V_TiPr_Codigo::TEXT,
"Se trabajo en la tabla de Titulos Profesionales (SAPW_Titulo_Profesional)",(SELECT
Current_TimeStamp));
        END IF;
    END IF;
    RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

```

```

--Función para el Mantenimiento de Observaciones para los Certificados
DROP FUNCTION SAPW_Mant_Observaciones(INTEGER,INTEGER, VARCHAR,
VARCHAR);
CREATE OR REPLACE FUNCTION
SAPW_Mant_Observaciones(INTEGER,INTEGER, VARCHAR, VARCHAR)
RETURNS INTEGER AS '
DECLARE
    V_Cab_Auditoria          ALIAS FOR $1;
    V_Obse_Codigo            ALIAS FOR $2;
    V_Obse_Descripcion        ALIAS FOR $3;
    V_Tipo                    ALIAS FOR $4;
    V_Secuencia               INTEGER;
BEGIN
    IF V_Tipo = "I" THEN
        SELECT NEXTVAL ("SAPW_S_Observaciones") INTO V_Secuencia;
    INSERT INTO SAPW_Observaciones VALUES(V_Secuencia, V_Obse_Descripcion);
    INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento

```

```

de Observaciones para los Certificados se Creo el registro con Código " ||
V_Secuencia::TEXT,
"Se trabajo en la tabla de Observaciones para los Certificados
(SAPW_Observaciones)",(SELECT Current_TimeStamp));
    ELSE
        V_Secuencia := V_Obse_Codigo;
        IF V_Tipo = "U" THEN
UPDATE SAPW_Observaciones SET Obse_Descripcion = V_Obse_Descripcion
WHERE Obse_Codigo = V_Obse_Codigo;
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Observaciones para los Certificados se Modificó el registro con Código " ||
V_Obse_Codigo::TEXT,
"Se trabajo en la tabla de Observaciones para los Certificados
(SAPW_Observaciones)",(SELECT Current_TimeStamp));
        ELSE
            IF V_Tipo = "D" THEN
DELETE FROM SAPW_Observaciones WHERE Obse_Codigo = V_Obse_Codigo;
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Observaciones para los Certificados se Eliminó el registro con Código " ||
V_Obse_Codigo::TEXT,
"Se trabajo en la tabla de Observaciones para los Certificados
(SAPW_Observaciones)",(SELECT Current_TimeStamp));
            END IF;
        END IF;
    END IF;
UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion = (SELECT
Current_Date);
    RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Laboratorios

```

DROP FUNCTION SAPW_Mant_Laboratorios(INTEGER,INTEGER, VARCHAR,
VARCHAR, INTEGER, VARCHAR);
CREATE OR REPLACE FUNCTION
SAPW_Mant_Laboratorios(INTEGER,INTEGER, VARCHAR, VARCHAR,
INTEGER, VARCHAR) RETURNS INTEGER AS '
    DECLARE
        V_Cab_Auditoria          ALIAS FOR $1;
        V_Labo_Codigo            ALIAS FOR $2;
        V_Labo_Descripcion       ALIAS FOR $3;
        V_Labo_Ubicacion         ALIAS FOR $4;
        V_Labo_Cupo             ALIAS FOR $5;
        V_Tipo                   ALIAS FOR $6;
        V_Secuencia              INTEGER;
    BEGIN
        IF V_Tipo = "I" THEN

```

```

SELECT NEXTVAL ("SAPW_S_Laboratorio") INTO V_Secuencia;
INSERT INTO SAPW_Laboratorio VALUES(V_Secuencia,
V_Labo_Descripcion, V_Labo_Ubicacion, V_Labo_Cupo);
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
Mantenimiento de Laboratorios se Creo el registro con Código " ||
V_Secuencia::TEXT,
"Se trabajo en la tabla de Laboratorios (SAPW_Laboratorio)",(SELECT
Current_TimeStamp));
ELSE
    V_Secuencia := V_Labo_Codigo;
    IF V_Tipo = "U" THEN
UPDATE SAPW_Laboratorio SET Labo_Descripcion = V_Labo_Descripcion,
Labo_Ubicacion = V_Labo_Ubicacion, Labo_Cupo = V_Labo_Cupo
WHERE Labo_Codigo = V_Labo_Codigo;
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Laboratorios se Modificó el registro con Código " || V_Labo_Codigo::TEXT,
"Se trabajo en la tabla de Laboratorios (SAPW_Laboratorio)",(SELECT
Current_TimeStamp));
    ELSE
        IF V_Tipo = "D" THEN
DELETE FROM SAPW_Laboratorio WHERE Labo_Codigo = V_Labo_Codigo;
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Laboratorios se Eliminó el registro con Código " || V_Labo_Codigo::TEXT,
"Se trabajo en la tabla de Laboratorios (SAPW_Laboratorio)",(SELECT
Current_TimeStamp));
        END IF;
    END IF;
END IF;
RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Aulas de Clase

```

DROP FUNCTION SAPW_Mant_Aulas(INTEGER,INTEGER, VARCHAR,
VARCHAR);
CREATE OR REPLACE FUNCTION SAPW_Mant_Aulas(INTEGER,INTEGER,
VARCHAR, VARCHAR) RETURNS INTEGER AS '
    DECLARE
        V_Cab_Auditoria          ALIAS FOR $1;
        V_SaCl_Codigo            ALIAS FOR $2;
        V_SaCl_Descripcion        ALIAS FOR $3;
        V_Tipo                    ALIAS FOR $4;
        V_Secuencia               INTEGER;
    BEGIN
        IF V_Tipo = "I" THEN
            SELECT NEXTVAL ("SAPW_S_Salones_Clase") INTO V_Secuencia;

```

```

INSERT INTO SAPW_Salon_Clase VALUES(V_Secuencia,
V_SaCl_Descripcion);
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Salones de Clase se Creo el registro con Código " || V_Secuencia::TEXT,
"Se trabajo en la tabla de Salones de Clase (SAPW_Salon_Clase)",(SELECT
Current_TimeStamp));
ELSE
V_Secuencia := V_SaCl_Codigo;
IF V_Tipo = "U" THEN
UPDATE SAPW_Salon_Clase SET SaCl_Descripcion = V_SaCl_Descripcion
WHERE SaCl_Codigo = V_SaCl_Codigo;
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Salones de Clase se Modificó el registro con Código " || V_SaCl_Codigo::TEXT,
"Se trabajo en la tabla de Salones de Clase (SAPW_Salon_Clase)",(SELECT
Current_TimeStamp));
ELSE
IF V_Tipo = "D" THEN
DELETE FROM SAPW_Salon_Clase WHERE SaCl_Codigo = V_SaCl_Codigo;
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
Mantenimiento de Salones de Clase se Eliminó el registro con Código " ||
V_SaCl_Codigo::TEXT,
"Se trabajo en la tabla de Salones de Clase (SAPW_Salon_Clase)",(SELECT
Current_TimeStamp));
END IF;
END IF;
END IF;
RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

```



```

--Función para el Mantenimiento de Politicas de Reservación de Laboratorios
DROP FUNCTION SAPW_Mant_PoliticasReservacion(INTEGER,INTEGER,
VARCHAR, VARCHAR);
CREATE OR REPLACE FUNCTION
SAPW_Mant_PoliticasReservacion(INTEGER,INTEGER, VARCHAR, VARCHAR)
RETURNS INTEGER AS '
DECLARE
V_Cab_Auditoria          ALIAS FOR $1;
V_PoRe_Codigo            ALIAS FOR $2;
V_PoRe_Descripción       ALIAS FOR $3;
V_Tipo                   ALIAS FOR $4;
V_Secuencia              INTEGER;
BEGIN
IF V_Tipo = "I" THEN
SELECT NEXTVAL ("SAPW_S_PoliticasReservacion") INTO V_Secuencia;

```



```

INSERT INTO SAPW_Politica_Reservacion VALUES(V_Secuencia,
V_PoRe_Descripcion);
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
Mantenimiento de Politicas de Reservación de Laboratorios se Creo el registro
con Código " || V_Secuencia::TEXT,
"Se trabajo en la tabla de Politicas de Reservación de Laboratorios
(SAPW_Reservacion_Laboratorio)",(SELECT Current_TimeStamp));
ELSE
    V_Secuencia := V_PoRe_Codigo;
    IF V_Tipo = "U" THEN
UPDATE SAPW_Politica_Reservacion SET PoRe_Descripcion =V_PoRe_Descripcion
WHERE PoRe_Codigo = V_PoRe_Codigo;
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Politicas de Reservación de Laboratorios se Modificó el registro con Código " ||
V_PoRe_Codigo::TEXT,
"Se trabajo en la tabla de Politicas de Reservación de Laboratorios
(SAPW_Politica_Reservacion)",(SELECT Current_TimeStamp));
ELSE
    IF V_Tipo = "D" THEN
DELETE FROM SAPW_Politica_Reservacion WHERE PoRe_Codigo =
V_PoRe_Codigo;
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Politicas de Reservación de Laboratorios se Eliminó el registro con Código " ||
V_PoRe_Codigo::TEXT,
"Se trabajo en la tabla de Politicas Reservación de Laboratorios
(SAPW_Politica_Reservacion)",(SELECT Current_TimeStamp));
END IF;
END IF;
END IF;
UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion = (SELECT
Current_Date);
RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Catalogo Digital

```

DROP FUNCTION SAPW_Mant_CatalogoDigital(INTEGER,INTEGER, VARCHAR,
DATE, VARCHAR, VARCHAR);
CREATE OR REPLACE FUNCTION
SAPW_Mant_CatalogoDigital(INTEGER,INTEGER, VARCHAR, DATE,
VARCHAR, VARCHAR) RETURNS INTEGER AS '
DECLARE
    V_Cab_Auditoria          ALIAS FOR $1;
    V_CaPr_Codigo            ALIAS FOR $2;
    V_CaPr_Descripcion        ALIAS FOR $3;
    V_CaPr_Fecha              ALIAS FOR $4;

```

```

V_CaPr_Url          ALIAS FOR $5;
V_Tipo              ALIAS FOR $6;
V_Secuencia         INTEGER;

BEGIN
  IF V_Tipo = "I" THEN
SELECT NEXTVAL ("SAPW_S_Catalogo_Protcom") INTO V_Secuencia;
    INSERT INTO SAPW_Catalogo_Protcom VALUES(V_Secuencia,
    V_CaPr_Descripcion, V_CaPr_Fecha, V_CaPr_Url);
    INSERT INTO SAPW_Detalle_Auditoria VALUES
    (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
    Mantenimiento de Catalogo Digital se Creo el registro con Código " ||
    V_Secuencia::TEXT,
    "Se trabajo en la tabla de Catalogo Digital
    (SAPW_Catalogo_Protcom)",(SELECT Current_TimeStamp));
    ELSE
      V_Secuencia := V_CaPr_Codigo;
      IF V_Tipo = "U" THEN
        UPDATE SAPW_Catalogo_Protcom SET CaPr_Descripcion =
        V_CaPr_Descripcion, CaPr_FechaPublicacion = V_CaPr_Fecha,
        CaPr_Url = V_CaPr_Url WHERE CaPr_Codigo = V_CaPr_Codigo;
        INSERT INTO SAPW_Detalle_Auditoria VALUES
        (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
        Mantenimiento de Catalogo Digital se Modificó el registro con Código "
        || V_CaPr_Codigo::TEXT,
        "Se trabajo en la tabla de Catalogo Digital
        (SAPW_Catalogo_Protcom)",(SELECT Current_TimeStamp));
        ELSE
          IF V_Tipo = "D" THEN
DELETE FROM SAPW_Catalogo_Protcom WHERE CaPr_Codigo = V_CaPr_Codigo;
            INSERT INTO SAPW_Detalle_Auditoria VALUES
            (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
            Mantenimiento de Catalogo Digital se Eliminó el registro con Código " ||
            V_CaPr_Codigo::TEXT,
            "Se trabajo en la tabla de Catalogo Digital
            (SAPW_Catalogo_Protcom)",(SELECT Current_TimeStamp));
            END IF;
          END IF;
        END IF;
      UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion = (SELECT
      Current_Date);
      RETURN V_Secuencia;
    END;
  ' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Foros de Discusión

```

DROP FUNCTION SAPW_Mant_Foros(INTEGER,INTEGER, VARCHAR, DATE,
VARCHAR);
CREATE OR REPLACE FUNCTION SAPW_Mant_Foros(INTEGER,INTEGER,
VARCHAR, DATE, VARCHAR) RETURNS INTEGER AS '

```



```

DECLARE
    V_Cab_Auditoria      ALIAS FOR $1;
    V_Foru_Codigo        ALIAS FOR $2;
    V_Foru_Nombre        ALIAS FOR $3;
    V_Foru_Fecha         ALIAS FOR $4;
    V_Tipo               ALIAS FOR $5;
    V_Secuencia          INTEGER;

BEGIN
    IF V_Tipo = "I" THEN
        SELECT NEXTVAL ("SAPW_S_Forum") INTO V_Secuencia;
        INSERT INTO SAPW_Forum VALUES(V_Secuencia,
V_Foru_Nombre, V_Foru_Fecha, 0);
        INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"
En el Mantenimiento de Forums se Creo el registro con Código "
|| V_Secuencia::TEXT,
"Se trabajo en la tabla de Forums (SAPW_Forum)",(SELECT
Current_TimeStamp));
    ELSE
        V_Secuencia := V_Foru_Codigo;
        IF V_Tipo = "U" THEN
            UPDATE SAPW_Forum SET Foru_Nombre = V_Foru_Nombre
            WHERE Foru_Numero = V_Foru_Codigo;
            INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
Mantenimiento de Forums se Modificó el registro con Código " ||
V_Foru_Codigo::TEXT,
"Se trabajo en la tabla de Forums (SAPW_Forum)",(SELECT
Current_TimeStamp));
        ELSE
            IF V_Tipo = "D" THEN
                DELETE FROM SAPW_Forum WHERE Foru_Numero = V_Foru_Codigo;
                INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
Mantenimiento de Forums se Eliminó el registro con Código " ||
V_Foru_Codigo::TEXT,
"Se trabajo en la tabla de Forums (SAPW_Forum)",(SELECT
Current_TimeStamp));
            END IF;
        END IF;
    END IF;
    RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Noticias T.I.

```

DROP FUNCTION SAPW_Mant_NoticiasTI(int4, int4, int4, varchar, varchar, varchar,
varchar, varchar, varchar);

```

```
CREATE FUNCTION public.sapw_mant_noticiasti(int4, int4, int4, varchar, varchar,
varchar, varchar, varchar, varchar) RETURNS int4 AS '
```

```
DECLARE
```

```
    V_Cab_Auditoria          ALIAS FOR $1;
    V_Link_Codigo            ALIAS FOR $2;
    V_Link_Tipo               ALIAS FOR $3;
    V_Link_Topico             ALIAS FOR $4;
    V_Link_Descripcion        ALIAS FOR $5;
    V_Link_Url                ALIAS FOR $6;
    V_Link_TipoLink           ALIAS FOR $7;
    V_Link_Campus             ALIAS FOR $8;
    V_Tipo                    ALIAS FOR $9;
    V_Secuencia               INTEGER;
```

```
BEGIN
```

```
    IF V_Tipo = "I" THEN
```

```
        SELECT NEXTVAL ("SAPW_S_Link") INTO V_Secuencia;
        INSERT INTO SAPW_Link VALUES(V_Secuencia,
        V_Link_Tipo, V_Link_Topico, V_Link_Descripcion, 0,
        V_Link_TipoLink, V_Link_Campus,
        NEXTVAL("SAPW_S_Foto_Cabecera"), V_Link_Url);
        INSERT INTO SAPW_Detalle_Auditoria VALUES
        (NEXTVAL("SAPW_S_Detalle_Auditoria"), V_Cab_Auditoria,
        En el Mantenimiento de Noticias T.I. se Creo el registro con
        Código " || V_Secuencia::TEXT, "Se trabajo en la tabla de
        Noticias T.I. (SAPW_Link)", (SELECT Current_TimeStamp));
```

```
    ELSE
```

```
        V_Secuencia := V_Link_Codigo;
```

```
        IF V_Tipo = "U" THEN
```

```
            UPDATE SAPW_Link SET Link_Descripcion =
            V_Link_Descripcion, Link_Topico = V_Link_Topico,
            Link_Url = V_Link_Url, Link_Tipo = V_Link_TipoLink,
            Tipo_Codigo = V_Link_Tipo, Link_Campus =
            V_Link_Campus WHERE Link_Codigo =
            V_Link_Codigo;
            INSERT INTO SAPW_Detalle_Auditoria VALUES
            (NEXTVAL("SAPW_S_Detalle_Auditoria"), V_Cab_Audi
            toria, "En el Mantenimiento de Noticias T.I. se Modificó
            el registro con Código " || V_Link_Codigo::TEXT, "Se
            trabajo en la tabla de Noticias T.I.
            (SAPW_Link)", (SELECT Current_TimeStamp));
```

```
        ELSE
```

```
            IF V_Tipo = "D" THEN
```

```
                DELETE FROM SAPW_Link WHERE
                Link_Codigo = V_Link_Codigo;
                INSERT INTO SAPW_Detalle_Auditoria
                VALUES
                (NEXTVAL("SAPW_S_Detalle_Auditoria"), V_Ca
                b_Auditoria, "En el Mantenimiento de Noticias T.I.
                se Eliminó el registro con Código " ||
                V_Link_Codigo::TEXT, "Se trabajo en la tabla de
```

```

        Noticias T.I. (SAPW_Link)",(SELECT
        Current_TimeStamp));
    END IF;
    END IF;
    END IF;
    UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
    (SELECT Current_Date);
    RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Recurso Humano

DROP FUNCTION

SAPW_Mant_RecursoHumano(INTEGER,INTEGER,VARCHAR,VARCHAR,VARCHAR,VARCHAR);

CREATE OR REPLACE FUNCTION

SAPW_Mant_RecursoHumano(INTEGER,INTEGER,VARCHAR,VARCHAR,VARCHAR,VARCHAR) RETURNS INTEGER AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_ReHu_Codigo	ALIAS FOR \$2;
V_ReHu_Encabezado	ALIAS FOR \$3;
V_ReHu_Conocimiento	ALIAS FOR \$4;
V_ReHu_Candidatos	ALIAS FOR \$5;
V_Tipo	ALIAS FOR \$6;
V_Secuencia	INTEGER;

BEGIN

IF V_Tipo = "I" THEN

SELECT NEXTVAL ("SAPW_S_Recurso_Humano") INTO V_Secuencia;

INSERT INTO SAPW_Recurso_Humano VALUES(V_Secuencia,
V_ReHu_Encabezado, V_ReHu_Conocimiento, V_ReHu_Candidatos);

INSERT INTO SAPW_Detalle_Auditoria VALUES

(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
Mantenimiento de Recursos Humanos se Creo el registro con Código " ||
V_Secuencia::TEXT,

"Se trabajo en la tabla de Recursos Humanos

(SAPW_Recurso_Humano)",(SELECT Current_TimeStamp));

ELSE

V_Secuencia := V_ReHu_Codigo;

IF V_Tipo = "U" THEN

UPDATE SAPW_Recurso_Humano SET ReHu_Encabezado =

V_ReHu_Encabezado, ReHu_Conocimiento = V_ReHu_Conocimiento,

ReHu_Candidatos = V_ReHu_Candidatos

WHERE ReHu_Codigo = V_ReHu_Codigo;

INSERT INTO SAPW_Detalle_Auditoria VALUES

(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
Mantenimiento de Recursos Humanos se Modificó el registro con Código " ||
V_ReHu_Codigo::TEXT,



```

"Se trabajo en la tabla de Recursos Humanos
(SAPW_Recurso_Humano)",(SELECT Current_TimeStamp));
      ELSE
        IF V_Tipo = "D" THEN
          DELETE FROM SAPW_Recurso_Humano WHERE ReHu_Codigo =
            V_ReHu_Codigo;
          INSERT INTO SAPW_Detalle_Auditoria VALUES
            (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
            Mantenimiento de Recursos Humanos se Eliminó el registro con Código " ||
            V_ReHu_Codigo::TEXT,
            "Se trabajo en la tabla de Recursos Humanos
            (SAPW_Recurso_Humano)",(SELECT Current_TimeStamp));
          END IF;
        END IF;
      END IF;
    UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion = (SELECT
      Current_Date);
    RETURN V_Secuencia;
  END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Solicitudes de Personal

DROP FUNCTION

SAPW_Mant_SolicitudPersonal(INTEGER,INTEGER,VARCHAR,VARCHAR,VARCHAR,DATE,VARCHAR,VARCHAR,VARCHAR);

CREATE OR REPLACE FUNCTION

SAPW_Mant_SolicitudPersonal(INTEGER,INTEGER,VARCHAR,VARCHAR,VARCHAR,DATE,VARCHAR,VARCHAR,VARCHAR) RETURNS INTEGER AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_SoPe_Codigo	ALIAS FOR \$2;
V_SoPe_Empresa	ALIAS FOR \$3;
V_SoPe_Ciudad	ALIAS FOR \$4;
V_SoPe_Requiere	ALIAS FOR \$5;
V_SoPe_Fecha	ALIAS FOR \$6;
V_SoPe_Telefono	ALIAS FOR \$7;
V_SoPe_Firma	ALIAS FOR \$8;
V_Tipo	ALIAS FOR \$9;
V_Secuencia	INTEGER;

BEGIN

```

  IF V_Tipo = "I" THEN
    SELECT NEXTVAL ("SAPW_S_Solicitud_Personal") INTO V_Secuencia;
    INSERT INTO SAPW_Solicitud_Personal VALUES(V_Secuencia,
      V_SoPe_Empresa, V_SoPe_Ciudad, V_SoPe_Requiere,
      V_SoPe_Fecha, V_SoPe_Telefono, V_SoPe_Firma);
    INSERT INTO SAPW_Detalle_Auditoria VALUES
      (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
      Mantenimiento de Solicitud de Personal se Creo el registro con Código " ||
      V_Secuencia::TEXT,

```

```

"Se trabajo en la tabla de Solicitud de Personal
(SAPW_Solicitud_Personal)",(SELECT Current_TimeStamp));
    ELSE
        V_Secuencia := V_SoPe_Codigo;
        IF V_Tipo = "U" THEN
            UPDATE SAPW_Solicitud_Personal SET SoPe_Empresa = V_SoPe_Empresa,
            SoPe_Ciudad = V_SoPe_Ciudad, SoPe_Requiere = V_SoPe_Requiere,
            SoPe_Telefono = V_SoPe_Telefono, SoPe_Firma = V_SoPe_Firma
            WHERE SoPe_Codigo = V_SoPe_Codigo;
            INSERT INTO SAPW_Detalle_Auditoria VALUES
            (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
            Mantenimiento de Solicitud de Personal se Modificó el registro con Código " ||
            V_SoPe_Codigo::TEXT,
            "Se trabajo en la tabla de Solicitud de Personal
            (SAPW_Solicitud_Personal)",(SELECT Current_TimeStamp));
            ELSE
                IF V_Tipo = "D" THEN
                    DELETE FROM SAPW_solicitud_Personal WHERE SoPe_Codigo =
                    V_SoPe_Codigo;
                    INSERT INTO SAPW_Detalle_Auditoria VALUES
                    (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
                    Mantenimiento de Solicitud de Personal se Eliminó el registro con Código " ||
                    V_SoPe_Codigo::TEXT,
                    "Se trabajo en la tabla de Solicitud de Personal
                    (SAPW_Solicitud_Personal)",(SELECT Current_TimeStamp));
                END IF;
            END IF;
        END IF;
        UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion = (SELECT
        Current_Date);
        RETURN V_Secuencia;
    END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Tip's Académicos

DROP FUNCTION

SAPW_Mant_TipsAcademicos(INTEGER,INTEGER,INTEGER,VARCHAR,VARCH
AR,VARCHAR,VARCHAR);

CREATE OR REPLACE FUNCTION

SAPW_Mant_TipsAcademicos(INTEGER,INTEGER,INTEGER,VARCHAR,VARCH
AR,VARCHAR,VARCHAR) RETURNS INTEGER AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_Tip_Numero	ALIAS FOR \$2;
V_Tip_Opcion	ALIAS FOR \$3;
V_Tip_Resumen	ALIAS FOR \$4;
V_Tip_Descripción	ALIAS FOR \$5;
V_Tip_Publicar	ALIAS FOR \$6;
V_Tipo	ALIAS FOR \$7;

```

V_Secuencia          INTEGER;
BEGIN
  IF V_Tipo = "I" THEN
    SELECT NEXTVAL ("SAPW_S_Tips") INTO V_Secuencia;
    INSERT INTO SAPW_Tip VALUES(V_Secuencia,
    V_Tip_Opcion, V_Tip_Resumen, V_Tip_Descripcion, 0,
    V_Tip_Publicar);
    INSERT INTO SAPW_Detalle_Auditoria VALUES
    (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"
    En el Mantenimiento de Tips Académicos se Creo el registro con
    Código " || V_Secuencia::TEXT,
    "Se trabajo en la tabla de Tips Académicos
    (SAPW_Tip)",(SELECT Current_TimeStamp));
  ELSE
    V_Secuencia := V_Tip_Numero;
    IF V_Tipo = "U" THEN
      UPDATE SAPW_Tip SET OpWe_Codigo = V_Tip_Opcion,
      Tip_Resumen = V_Tip_Resumen, Tip_Descripcion =
      V_Tip_Descripcion, Tip_Publicar = V_Tip_Publicar WHERE
      Tip_Numero = V_Tip_Numero;
      INSERT INTO SAPW_Detalle_Auditoria VALUES
      (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
      Mantenimiento de Tips Académicos se Modificó el registro con Código "
      || V_Tip_Numero::TEXT,
      "Se trabajo en la tabla de Tips Académicos (SAPW_Tip)",(SELECT
      Current_TimeStamp));
    ELSE
      IF V_Tipo = "D" THEN
        DELETE FROM SAPW_Tip WHERE Tip_Numero = V_Tip_Numero;
        INSERT INTO SAPW_Detalle_Auditoria VALUES
        (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
        Mantenimiento de Tips Académicos se Eliminó el registro con Código " ||
        V_Tip_Numero::TEXT,
        "Se trabajo en la tabla de Tips Académicos (SAPW_Tip)",(SELECT
        Current_TimeStamp));
      END IF;
    END IF;
  END IF;
  UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
  (SELECT Current_Date);
  RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Departamentos

DROP FUNCTION

SAPW_Mant_Departamentos(INTEGER,INTEGER,VARCHAR,VARCHAR,VARCH
AR,INTEGER,VARCHAR);

CREATE OR REPLACE FUNCTION

SAPW_Mant_Departamentos(INTEGER,INTEGER,VARCHAR,VARCHAR,VARCHAR,INTEGER,VARCHAR) RETURNS INTEGER AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_Depa_Codigo	ALIAS FOR \$2;
V_Depa_Jefe	ALIAS FOR \$3;
V_Depa_Nombre	ALIAS FOR \$4;
V_Depa_Descripcion	ALIAS FOR \$5;
V_Depa_Extension	ALIAS FOR \$6;
V_Tipo	ALIAS FOR \$7;
V_Secuencia	INTEGER;

BEGIN

IF V_Tipo = "I" THEN

SELECT NEXTVAL ("SAPW_S_Departamento") INTO V_Secuencia;
 INSERT INTO SAPW_Departamento VALUES(V_Secuencia,
 V_Depa_Jefe, V_Depa_Nombre, V_Depa_Descripcion, 0,
 V_Depa_Extension);

INSERT INTO SAPW_Detalle_Auditoria VALUES
 (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
 Mantenimiento de Departamentos se Creo el registro con Código " ||
 V_Secuencia::TEXT,

"Se trabajo en la tabla de Departamentos

(SAPW_Departamento)",(SELECT Current_TimeStamp));

ELSE

V_Secuencia := V_Depa_Codigo;

IF V_Tipo = "U" THEN

UPDATE SAPW_Departamento SET Pers_Cedula =

V_Depa_Jefe, Depa_Nombre = V_Depa_Nombre,

Depa_Descripcion = V_Depa_Descripcion,

Depa_Extension = V_Depa_Extension

WHERE Depa_Codigo = V_Depa_Codigo;

INSERT INTO SAPW_Detalle_Auditoria VALUES

(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Audi
 toria,"En el Mantenimiento de Departamentos se Modificó
 el registro con Código " || V_Depa_Codigo::TEXT,

"Se trabajo en la tabla de Departamentos

(SAPW_Departamento)",(SELECT
 Current_TimeStamp));

ELSE

IF V_Tipo = "D" THEN

DELETE FROM SAPW_Departamento WHERE

Depa_Codigo = V_Depa_Codigo;

INSERT INTO SAPW_Detalle_Auditoria VALUES

(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Audi
 toria,"En el Mantenimiento de Departamentos se Eliminó
 el registro con Código " || V_Depa_Codigo::TEXT,

"Se trabajo en la tabla de Departamentos

(SAPW_Departamento)",(SELECT
 Current_TimeStamp));

```

        END IF;
    END IF;
END IF;
UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion = (SELECT
Current_Date);
RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Eventos

DROP FUNCTION

SAPW_Mant_Eventos(INTEGER,INTEGER,VARCHAR,DATE,VARCHAR,INTEGER, VARCHAR);

CREATE OR REPLACE FUNCTION

SAPW_Mant_Eventos(INTEGER,INTEGER,VARCHAR,DATE,VARCHAR,INTEGER, VARCHAR) RETURNS INTEGER AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_Even_Codigo	ALIAS FOR \$2;
V_Even_Descripcion	ALIAS FOR \$3;
V_Even_Fecha	ALIAS FOR \$4;
V_Even_Lugar	ALIAS FOR \$5;
V_Even_Estado	ALIAS FOR \$6;
V_Tipo	ALIAS FOR \$7;
V_Secuencia	INTEGER;

BEGIN

IF V_Tipo = "I" THEN

```

        SELECT NEXTVAL ("SAPW_S_Evento") INTO V_Secuencia;
        INSERT INTO SAPW_Evento VALUES(V_Secuencia,
V_Even_Descripcion, V_Even_Fecha, V_Even_Lugar, V_Even_Estado);
        INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Eventos se Creo el registro con Código " || V_Secuencia::TEXT,
"Se trabajo en la tabla de Eventos
(SAPW_Evento)",(SELECT Current_TimeStamp));

```

ELSE

```

        V_Secuencia := V_Even_Codigo;
        IF V_Tipo = "U" THEN
            UPDATE SAPW_Evento SET Even_Lugar =
V_Even_Lugar, Even_Fecha = V_Even_Fecha,
Even_Descripcion =
V_Even_Descripcion, Esta_Codigo = V_Even_Estado
WHERE Even_Codigo =
V_Even_Codigo;

```

```

            INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Eventos se Modificó el registro con Código " || V_Even_Codigo::TEXT,
"Se trabajo en la tabla de Eventos
(SAPW_Evento)",(SELECT Current_TimeStamp));

```

```

ELSE
    IF V_Tipo = "D" THEN
        DELETE FROM SAPW_Evento WHERE
Even_Codigo = V_Even_Codigo;
        INSERT INTO SAPW_Detalle_Auditoria
VALUES (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
Mantenimiento de Eventos se Eliminó el registro con Código " ||
V_Even_Codigo::TEXT,
                                "Se trabajo en la tabla de Eventos
(SAPW_Evento)",(SELECT Current_TimeStamp));
        END IF;
    END IF;
END IF;
UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
(SELECT Current_Date);
RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Cartera de Clientes

```

DROP FUNCTION
SAPW_Mant_Clientes(INTEGER,INTEGER,INTEGER,VARCHAR,VARCHAR,VA
RCHAR,VARCHAR,VARCHAR,VARCHAR);
CREATE OR REPLACE FUNCTION
SAPW_Mant_Clientes(INTEGER,INTEGER,INTEGER,VARCHAR,VARCHAR,VA
RCHAR,VARCHAR,VARCHAR,VARCHAR) RETURNS INTEGER AS '
DECLARE
    V_Cab_Auditoria           ALIAS FOR $1;
    V_Clie_Codigo             ALIAS FOR $2;
    V_Clie_Tipo               ALIAS FOR $3;
    V_Clie_Nombre             ALIAS FOR $4;
    V_Clie_Direccion          ALIAS FOR $5;
    V_Clie_Telefono           ALIAS FOR $6;
    V_Clie_Email              ALIAS FOR $7;
    V_Clie_Url                ALIAS FOR $8;
    V_Tipo                   ALIAS FOR $9;
    V_Secuencia               INTEGER;
BEGIN
    IF V_Tipo = "I" THEN
        SELECT NEXTVAL ("SAPW_S_Clientes") INTO V_Secuencia;
        INSERT INTO SAPW_Cliente VALUES(V_Secuencia,
V_Clie_Tipo, V_Clie_Nombre, V_Clie_Direccion, V_Clie_Telefono,
                                V_Clie_Email, V_Clie_Url);
        INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Cartera de Clientes se Creo el registro con Código " || V_Secuencia::TEXT,
                                "Se trabajo en la tabla de Cartera de Clientes
(SAPW_Cliente)",(SELECT Current_TimeStamp));
    ELSE

```

```

V_Secuencia := V_Clie_Codigo;
IF V_Tipo = "U" THEN
    UPDATE SAPW_Cliente SET Tipo_Codigo =
V_Clie_Tipo, Clie_Nombre = V_Clie_Nombre,
Clie_Direccion = V_Clie_Direccion,
Clie_Telefono = V_Clie_Telefono,
Clie_Email = V_Clie_Email,
Clie_Url = V_Clie_Url
    WHERE Clie_Codigo = V_Clie_Codigo;
    INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"), V_Cab_Auditoria, "En el Mantenimiento
de Cartera de Clientes se Modificó el registro con Código " || V_Clie_Codigo::TEXT,
"Se trabajo en la tabla de Cartera de Clientes (SAPW_Cliente)", (SELECT
Current_TimeStamp));
ELSE
    IF V_Tipo = "D" THEN
        DELETE FROM SAPW_Cliente WHERE
Clie_Codigo = V_Clie_Codigo;
        INSERT INTO SAPW_Detalle_Auditoria
VALUES (NEXTVAL("SAPW_S_Detalle_Auditoria"), V_Cab_Auditoria, "En el
Mantenimiento de Cartera de Clientes se Eliminó el registro con Código " ||
V_Clie_Codigo::TEXT,
"Se trabajo en la tabla de Cartera de
Clientes (SAPW_Cliente)", (SELECT Current_TimeStamp));
    END IF;
END IF;
END IF;
RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

```



--Función para el Mantenimiento de Diccionario de Búsqueda del Portal

DROP FUNCTION

SAPW_Mant_Diccionario(INTEGER, INTEGER, VARCHAR, VARCHAR, VARCHAR,
VARCHAR, INTEGER, VARCHAR);

CREATE OR REPLACE FUNCTION

SAPW_Mant_Diccionario(INTEGER, INTEGER, VARCHAR, VARCHAR, VARCHAR,
VARCHAR, INTEGER, VARCHAR) RETURNS INTEGER AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_Dicc_Codigo	ALIAS FOR \$2;
V_Dicc_Titulo	ALIAS FOR \$3;
V_Dicc_Frase	ALIAS FOR \$4;
V_Dicc_Comentario	ALIAS FOR \$5;
V_Dicc_Url	ALIAS FOR \$6;
V_Dicc_Estado	ALIAS FOR \$7;
V_Tipo	ALIAS FOR \$8;
V_Secuencia	INTEGER;

BEGIN

```

        IF V_Tipo = "I" THEN
            SELECT NEXTVAL ("SAPW_S_Diccionarios") INTO
V_Secuencia;
            INSERT INTO SAPW_Diccionario VALUES(V_Secuencia,
V_Dicc_Titulo, V_Dicc_Frase, V_Dicc_Comentario,
V_Dicc_Url, 0, V_Dicc_Estado);
            INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Diccionario de Búsqueda del Portal se Creo el registro con Código " ||
V_Secuencia::TEXT,
"Se trabajo en la tabla de Diccionario de Buqueda del
Portal (SAPW_Diccionario)",(SELECT Current_TimeStamp));
        ELSE
            V_Secuencia := V_Dicc_Codigo;
            IF V_Tipo = "U" THEN
                UPDATE SAPW_Diccionario SET Dicc_Titulo =
V_Dicc_Titulo, Dicc_Frase = V_Dicc_Frase, Dicc_Comentario = V_Dicc_Comentario,
Dicc_Url = V_Dicc_Url,
Esta_Codigo = V_Dicc_Estado
WHERE Dicc_Codigo =
V_Dicc_Codigo;
                INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Diccionario de Búsqueda del Portal se Modificó el registro con Código " ||
V_Dicc_Codigo::TEXT,
"Se trabajo en la tabla de Diccionario de
Búsqueda del Portal (SAPW_Diccionario)",(SELECT Current_TimeStamp));
            ELSE
                IF V_Tipo = "D" THEN
                    DELETE FROM SAPW_Diccionario WHERE
Dicc_Codigo = V_Dicc_Codigo;
                    INSERT INTO SAPW_Detalle_Auditoria
VALUES (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
Mantenimiento de Diccionario de Búsqueda del Portal se Eliminó el registro con
Código " || V_Dicc_Codigo::TEXT,
"Se trabajo en la tabla de Diccionario
de Búsqueda del Portal (SAPW_Diccionario)",(SELECT Current_TimeStamp));
                END IF;
            END IF;
        END IF;
        RETURN V_Secuencia;
    END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Años

```

DROP FUNCTION SAPW_Mant_Años(INTEGER,INTEGER, INTEGER,
VARCHAR);
CREATE OR REPLACE FUNCTION SAPW_Mant_Años(INTEGER,INTEGER,
INTEGER, VARCHAR) RETURNS INTEGER AS '

```

```

DECLARE
    V_Cab_Auditoria          ALIAS FOR $1;
    V_Anio_Codigo            ALIAS FOR $2;
    V_Anio_Descripcion       ALIAS FOR $3;
    V_Tipo                   ALIAS FOR $4;
    V_Secuencia              INTEGER;
BEGIN
    IF V_Tipo = "I" THEN
        SELECT NEXTVAL ("SAPW_S_Anio") INTO V_Secuencia;
        INSERT INTO SAPW_Anio VALUES(V_Secuencia,
V_Anio_Descripcion);
        INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Años se Creo el registro con Código " || V_Secuencia::TEXT,
"Se trabajo en la tabla de Años
(SAPW_Anio)",(SELECT Current_TimeStamp));
    ELSE
        V_Secuencia := V_Anio_Codigo;
        IF V_Tipo = "U" THEN
            UPDATE SAPW_Anio SET Anio_Descripcion =
V_Anio_Descripcion
                                WHERE Anio_Codigo = V_Anio_Codigo;
            INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Años se Modificó el registro con Código " || V_Anio_Codigo::TEXT,
"Se trabajo en la tabla de Años
(SAPW_Anio)",(SELECT Current_TimeStamp));
        ELSE
            IF V_Tipo = "D" THEN
                DELETE FROM SAPW_Anio WHERE
Anio_Codigo = V_Anio_Codigo;
                INSERT INTO SAPW_Detalle_Auditoria
VALUES (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
Mantenimiento de Años se Eliminó el registro con Código " || V_Anio_Codigo::TEXT,
"Se trabajo en la tabla de Años
(SAPW_Anio)",(SELECT Current_TimeStamp));
            END IF;
        END IF;
    END IF;
    RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Horarios

```

DROP FUNCTION SAPW_Mant_Horarios(INTEGER,INTEGER, TIME, TIME,
VARCHAR);
CREATE OR REPLACE FUNCTION SAPW_Mant_Horarios(INTEGER,INTEGER,
TIME, TIME, VARCHAR) RETURNS INTEGER AS '
DECLARE

```



```

V_Cab_Auditoria      ALIAS FOR $1;
V_Hora_Codigo        ALIAS FOR $2;
V_Hora_Inicio        ALIAS FOR $3;
V_Hora_Fin           ALIAS FOR $4;
V_Tipo               ALIAS FOR $5;
V_Secuencia          INTEGER;

BEGIN
  IF V_Tipo = "I" THEN
    SELECT NEXTVAL ("SAPW_S_Horario") INTO V_Secuencia;
    INSERT INTO SAPW_Horario VALUES(V_Secuencia,
V_Hora_Inicio, V_Hora_Fin);
    INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Horarios se Creo el registro con Código " || V_Secuencia::TEXT,
"Se trabajo en la tabla de Horarios
(SAPW_Horario)",(SELECT Current_TimeStamp));
  ELSE
    V_Secuencia := V_Hora_Codigo;
    IF V_Tipo = "U" THEN
      UPDATE SAPW_Horario SET Hora_Inicio =
V_Hora_Inicio, Hora_Fin = V_Hora_Fin
      WHERE Hora_Codigo =
V_Hora_Codigo;
      INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Horarios se Modificó el registro con Código " || V_Hora_Codigo::TEXT,
"Se trabajo en la tabla de Horarios
(SAPW_Horario)",(SELECT Current_TimeStamp));
    ELSE
      IF V_Tipo = "D" THEN
        DELETE FROM SAPW_Horario WHERE
Hora_Codigo = V_Hora_Codigo;
        INSERT INTO SAPW_Detalle_Auditoria
VALUES (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
Mantenimiento de Horarios se Eliminó el registro con Código " ||
V_Hora_Codigo::TEXT,
"Se trabajo en la tabla de Horarios
(SAPW_Anio)",(SELECT Current_TimeStamp));
      END IF;
    END IF;
  END IF;
  RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

--Función para el Mantenimiento de Materias
DROP FUNCTION
SAPW_Mant_Materias(INTEGER,VARCHAR,INTEGER,VARCHAR,VARCHAR,V
ARCHAR);

```

CREATE OR REPLACE FUNCTION

SAPW_Mant_Materias(INTEGER,VARCHAR,INTEGER,VARCHAR,VARCHAR,VARCHAR) RETURNS VARCHAR AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_Mate_Codigo	ALIAS FOR \$2;
V_Mate_Tipo	ALIAS FOR \$3;
V_Mate_Nombre	ALIAS FOR \$4;
V_Mate_Descripcion	ALIAS FOR \$5;
V_Tipo	ALIAS FOR \$6;

BEGIN

IF V_Tipo = "I" THEN

INSERT INTO SAPW_Materia VALUES(V_Mate_Codigo,
V_Mate_Tipo, V_Mate_Nombre, V_Mate_Descripcion);

INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Materias se Creo el registro con Código " || V_Mate_Codigo::TEXT,
"Se trabajo en la tabla de Materias
(SAPW_Materia)",(SELECT Current_TimeStamp));

ELSE

IF V_Tipo = "U" THEN

UPDATE SAPW_Materia SET Tipo_Codigo =
V_Mate_Tipo, Mate_Nombre = V_Mate_Nombre, Mate_Descripcion =
V_Mate_Descripcion

WHERE Mate_Codigo =

V_Mate_Codigo;

INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Materias se Modificó el registro con Código " || V_Mate_Codigo::TEXT,
"Se trabajo en la tabla de Materias
(SAPW_Materia)",(SELECT Current_TimeStamp));

ELSE

IF V_Tipo = "D" THEN

DELETE FROM SAPW_Materia WHERE
Mate_Codigo = V_Mate_Codigo;

INSERT INTO SAPW_Detalle_Auditoria
VALUES (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
Mantenimiento de Materias se Eliminó el registro con Código " ||
V_Mate_Codigo::TEXT,

"Se trabajo en la tabla de Materias
(SAPW_Materia)",(SELECT Current_TimeStamp));

END IF;

END IF;

END IF;

RETURN V_Mate_Codigo;

END;

' LANGUAGE 'plpgsql';

--Función para el Mantenimiento de Programoteca

DROP FUNCTION

SAPW_Mant_Programoteca(INTEGER,INTEGER,INTEGER,VARCHAR,VARCHAR,
VARCHAR,VARCHAR,VARCHAR,VARCHAR);

CREATE OR REPLACE FUNCTION

SAPW_Mant_Programoteca(INTEGER,INTEGER,INTEGER,VARCHAR,VARCHAR,
VARCHAR,VARCHAR,VARCHAR,VARCHAR) RETURNS INTEGER AS '

DECLARE

V_Cab_Auditoria ALIAS FOR \$1;

V_CoFu_Codigo ALIAS FOR \$2;

V_CoFu_Estado ALIAS FOR \$3;

V_CoFu_Usuario ALIAS FOR \$4;

V_CoFu_Ruta ALIAS FOR \$5;

V_CoFu_Titulo ALIAS FOR \$6;

V_CoFu_Lenguaje ALIAS FOR \$7;

V_CoFu_Descripción ALIAS FOR \$8;

V_Tipo ALIAS FOR \$9;

V_Secuencia INTEGER;

BEGIN

IF V_Tipo = "I" THEN

SELECT NEXTVAL ("SAPW_S_Codigo_Fuente") INTO

V_Secuencia;

INSERT INTO SAPW_Codigo_Fuente VALUES(V_Secuencia,

V_CoFu_Estado, V_CoFu_Usuario, V_CoFu_Ruta, V_CoFu_Titulo,

V_CoFu_Lenguaje, 0, 0, 0,

V_CoFu_Descripcion);

INSERT INTO SAPW_Detalle_Auditoria VALUES

(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Programoteca se Creo el registro con Código " || V_Secuencia::TEXT,

"Se trabajo en la tabla de Programoteca

(SAPW_Codigo_Fuente)",(SELECT Current_TimeStamp));

ELSE

V_Secuencia := V_CoFu_Codigo;

IF V_Tipo = "U" THEN

UPDATE SAPW_Codigo_Fuente SET Esta_Codigo =

V_CoFu_Estado, CoFu_Ruta = V_CoFu_Ruta, CoFu_Titulo = V_CoFu_Titulo,

CoFu_Lenguaje =

V_CoFu_Lenguaje, CoFu_Descripcion = V_CoFu_Descripcion

WHERE CoFu_Codigo =

V_CoFu_Codigo;

INSERT INTO SAPW_Detalle_Auditoria VALUES

(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Programoteca se Modificó el registro con Código " || V_CoFu_Codigo::TEXT,

"Se trabajo en la tabla de Programoteca

(SAPW_Codigo_Fuente)",(SELECT Current_TimeStamp));

ELSE

IF V_Tipo = "D" THEN

DELETE FROM SAPW_Codigo_Fuente WHERE

CoFu_Codigo = V_CoFu_Codigo;

```

INSERT INTO SAPW_Detalle_Auditoria
VALUES (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
Mantenimiento de Programoteca se Eliminó el registro con Código " ||
V_CoFu_Codigo::TEXT,
"Se trabajo en la tabla de Programoteca
(SAPW_Codigo_Fuente)",(SELECT Current_TimeStamp));
END IF;
END IF;
END IF;
UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
(SELECT Current_Date);
RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Carreras Universitarias

```

DROP FUNCTION SAPW_Mant_CarrerasUniversitarias(int4, int4, int4, varchar,
varchar, varchar, varchar, varchar, varchar, varchar, varchar)
CREATE FUNCTION public.sapw_mant_carrerasuniversitarias(int4, int4, int4,
varchar, varchar, varchar, varchar, varchar, varchar, varchar, varchar) RETURNS int4
AS '

```

```

DECLARE

```

```

    V_Cab_Auditoria          ALIAS FOR $1;
    V_Carr_Codigo            ALIAS FOR $2;
    V_Carr_Tipo              ALIAS FOR $3;
    V_Carr_Nombre            ALIAS FOR $4;
    V_Carr_Descripcion       ALIAS FOR $5;
    V_Carr_Titulo            ALIAS FOR $6;
    V_Carr_Duracion          ALIAS FOR $7;
    V_Carr_Flujo             ALIAS FOR $8;
    V_Carr_Pagina            ALIAS FOR $9;
    V_Carr_Thumbnail         ALIAS FOR $10;
    V_Tipo                   ALIAS FOR $11;
    V_Secuencia              INTEGER;

```

```

BEGIN

```

```

    IF V_Tipo = "I" THEN

```

```

        SELECT NEXTVAL ("SAPW_S_Carreras") INTO V_Secuencia;
        INSERT INTO SAPW_Carrera VALUES(V_Secuencia,
V_Carr_Tipo, V_Carr_Nombre, V_Carr_Descripcion, V_Carr_Titulo,
V_Carr_Duracion, V_Carr_Flujo,
V_Carr_Pagina,V_Carr_Thumbnail);

```

```

        INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Carreras Universitarias se Creo el registro con Código " || V_Secuencia::TEXT,
"Se trabajo en la tabla de Carreras Universitarias
(SAPW_Carrera)",(SELECT Current_TimeStamp));

```

```

    ELSE

```

```

        V_Secuencia := V_Carr_Codigo;
        IF V_Tipo = "U" THEN

```

```

UPDATE SAPW_Carrera SET Tipo_Codigo =
V_Carr_Tipo, Carr_Nombre = V_Carr_Nombre, Carr_Titulo = V_Carr_Titulo,
Carr_Duracion = V_Carr_Duracion,
Carr_Descripcion = V_Carr_Descripcion, Carr_Flujo = V_Carr_Flujo, Carr_Pagina =
V_Carr_Pagina, Carr_Imagen = V_Carr_Thumbnail
WHERE Carr_Codigo =
V_Carr_Codigo;

INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Carreras Universitarias se Modificó el registro con Código " ||
V_Carr_Codigo::TEXT,
"Se trabajo en la tabla de Carreras
Universitarias (SAPW_Carrera)",(SELECT Current_TimeStamp));
ELSE
IF V_Tipo = "D" THEN
DELETE FROM SAPW_Carrera WHERE
Carr_Codigo = V_Carr_Codigo;
INSERT INTO SAPW_Detalle_Auditoria
VALUES (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
Mantenimiento de Carreras Universitarias se Eliminó el registro con Código " ||
V_Carr_Codigo::TEXT,
"Se trabajo en la tabla de Carreras
Universitarias (SAPW_Carrera)",(SELECT Current_TimeStamp));
END IF;
END IF;
END IF;
UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
(SELECT Current_Date);
RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

```

```

--Función para el Mantenimiento de Perfil Empresarial por Carreras Universitarias
DROP FUNCTION
SAPW_Mant_PerfilEmpresarial(INTEGER,INTEGER,INTEGER,INTEGER,VARCH
AR,VARCHAR);
CREATE OR REPLACE FUNCTION
SAPW_Mant_PerfilEmpresarial(INTEGER,INTEGER,INTEGER,INTEGER,VARCH
AR,VARCHAR) RETURNS INTEGER AS '
DECLARE
    V_Cab_Auditoria          ALIAS FOR $1;
    V_PeEm_Codigo            ALIAS FOR $2;
    V_PeEm_Carrera           ALIAS FOR $3;
    V_PeEm_Tipo              ALIAS FOR $4;
    V_PeEm_Descripcion       ALIAS FOR $5;
    V_Tipo                   ALIAS FOR $6;
    V_Secuencia              INTEGER;

```



```

BEGIN
    IF V_Tipo = "I" THEN
        SELECT NEXTVAL ("SAPW_S_Perfil_Empresarial") INTO
V_Secuencia;
        INSERT INTO SAPW_Perfil_Empresarial
VALUES(V_Secuencia, V_PeEm_Carrera, V_PeEm_Tipo, V_PeEm_Descripcion);
        INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"), V_Cab_Auditoria, "En el Mantenimiento
de Perfil Empresarial de la Carrera Universitaria se Creo el registro con Código " ||
V_Secuencia::TEXT || " Perteneiente a la Carrera Universitaria con Código " ||
V_PeEm_Carrera::TEXT,
        "Se trabajo en la tabla de Perfil Empresarial de la
Carrera Universitaria (SAPW_Perfil_Empresarial)", (SELECT Current_TimeStamp));
    ELSE
        V_Secuencia := V_PeEm_Codigo;
        IF V_Tipo = "U" THEN
            UPDATE SAPW_Perfil_Empresarial SET Tipo_Codigo =
V_PeEm_Tipo, PeEm_Descripcion = V_PeEm_Descripcion
        WHERE Carr_Codigo =
V_PeEm_Carrera AND PeEm_Codigo = V_PeEm_Codigo;
            INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"), V_Cab_Auditoria, "En el Mantenimiento
de Perfil Empresarial de la Carrera Universitaria se Modificó el registro con Código " ||
V_PeEm_Codigo::TEXT || " Perteneiente a la Carrera Universitaria con Código " ||
V_PeEm_Carrera::TEXT,
        "Se trabajo en la tabla de Perfil Empresarial de
la Carrera Universitaria (SAPW_Perfil_Empresarial)", (SELECT Current_TimeStamp));
        ELSE
            IF V_Tipo = "D" THEN
                DELETE FROM SAPW_Perfil_Empresarial
WHERE Carr_Codigo = V_PeEm_Carrera AND PeEm_Codigo = V_PeEm_Codigo;
                INSERT INTO SAPW_Detalle_Auditoria
VALUES (NEXTVAL("SAPW_S_Detalle_Auditoria"), V_Cab_Auditoria, "En el
Mantenimiento de Perfil Empresarial de la Carrera Universitaria se Eliminó el registro
con Código " || V_PeEm_Codigo::TEXT || " Perteneiente a la Carrera Universitaria con
Código " || V_PeEm_Carrera::TEXT,
        "Se trabajo en la tabla de Perfil
Empresarial de la Carrera Universitaria (SAPW_Perfil_Empresarial)", (SELECT
Current_TimeStamp));
            END IF;
        END IF;
    END IF;
    UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
(SELECT Current_Date);
    RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

```



--Función para el Mantenimiento de Niveles Académicos de las Carreras Universitarias
DROP FUNCTION

SAPW_Mant_NivelAcademico(INTEGER,INTEGER,INTEGER,VARCHAR,VARCHAR);

CREATE OR REPLACE FUNCTION

SAPW_Mant_NivelAcademico(INTEGER,INTEGER,INTEGER,VARCHAR,VARCHAR) RETURNS INTEGER AS '

DECLARE

V_Cab_Auditoria ALIAS FOR \$1;

V_NiAc_Codigo ALIAS FOR \$2;

V_NiAc_Carrera ALIAS FOR \$3;

V_NiAc_Descripcion ALIAS FOR \$4;

V_Tipo ALIAS FOR \$5;

V_Secuencia INTEGER;

BEGIN

IF V_Tipo = "I" THEN

SELECT NEXTVAL ("SAPW_S_Nivel_Academico") INTO

V_Secuencia;

INSERT INTO SAPW_Nivel_Academico

VALUES(V_Secuencia, V_NiAc_Carrera, V_NiAc_Descripcion);

INSERT INTO SAPW_Detalle_Auditoria VALUES

(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento de Niveles Académicos de la Carrera Universitaria se Creo el registro con Código " ||

V_Secuencia::TEXT || " Perteneiente a la Carrera Universitaria con Código " ||

V_NiAc_Carrera::TEXT,

"Se trabajo en la tabla de Niveles Académicos de la Carrera Universitaria (SAPW_Nivel_Academico)",(SELECT Current_TimeStamp));

ELSE

V_Secuencia := V_NiAc_Codigo;

IF V_Tipo = "U" THEN

UPDATE SAPW_Nivel_Academico SET

NiAc_Descripcion = V_NiAc_Descripcion

WHERE Carr_Codigo =

V_NiAc_Carrera AND NiAc_Codigo = V_NiAc_Codigo;

INSERT INTO SAPW_Detalle_Auditoria VALUES

(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento de Niveles Académicos de la Carrera Universitaria se Modificó el registro con Código " ||

|| V_NiAc_Codigo::TEXT || " Perteneiente a la Carrera Universitaria con Código " ||

V_NiAc_Carrera::TEXT,

"Se trabajo en la tabla de Niveles Académicos de la Carrera Universitaria (SAPW_Nivel_Academico)",(SELECT Current_TimeStamp));

ELSE

IF V_Tipo = "D" THEN

DELETE FROM SAPW_Nivel_Academico

WHERE Carr_Codigo = V_NiAc_Carrera AND NiAc_Codigo = V_NiAc_Codigo;

INSERT INTO SAPW_Detalle_Auditoria

VALUES (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento de Niveles Académicos de la Carrera Universitaria se Eliminó el

```

registro con Código " || V_NiAc_Codigo::TEXT || " Perteneciente a la Carrera
Universitaria con Código " || V_NiAc_Carrera::TEXT,
                                "Se trabajo en la tabla de Niveles
Académicos de la Carrera Universitaria (SAPW_Nivel_Academico)",(SELECT
Current_TimeStamp));
                                END IF;
                                END IF;
                                END IF;
                                UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
(SELECT Current_Date);
                                RETURN V_Secuencia;
                                END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Pensum Académico

DROP FUNCTION

SAPW_Mant_PensumAcademico(INTEGER,VARCHAR,INTEGER,INTEGER,VAR
CHAR);

CREATE OR REPLACE FUNCTION

SAPW_Mant_PensumAcademico(INTEGER,VARCHAR,INTEGER,INTEGER,VAR
CHAR) RETURNS VARCHAR AS '

DECLARE

V_Cab_Auditoria ALIAS FOR \$1;

V_PeAc_Materia ALIAS FOR \$2;

V_PeAc_Nivel ALIAS FOR \$3;

V_PeAc_Carrera ALIAS FOR \$4;

V_Tipo ALIAS FOR \$5;

BEGIN

IF V_Tipo = "I" THEN

INSERT INTO SAPW_Materia_Nivel

VALUES(V_PeAc_Materia, V_PeAc_Nivel, V_PeAc_Carrera);

INSERT INTO SAPW_Detalle_Auditoria VALUES

(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Pensum Académico se agrego la Materia con Código " || V_PeAc_Materia || " al
Nivel " || V_PeAc_Nivel::TEXT || " Perteneciente a la Carrera Universitaria con Código
" || V_PeAc_Carrera::TEXT, "Se trabajo en la tabla de Pensum Académico
(SAPW_Materia_Nivel)",(SELECT Current_TimeStamp));

ELSE

IF V_Tipo = "D" THEN

DELETE FROM SAPW_Materia_Nivel WHERE

Carr_Codigo = V_PeAc_Carrera AND NiAc_Codigo = V_PeAc_Nivel AND

Mate_Codigo = V_PeAc_Materia;

INSERT INTO SAPW_Detalle_Auditoria VALUES

(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Pensum Académico se Eliminó la Materia con Código " || V_PeAc_Materia || " al
Nivel " || V_PeAc_Nivel::TEXT || " Perteneciente a la Carrera Universitaria con Código
" || V_PeAc_Carrera::TEXT, "Se trabajo en la tabla de Pensum Académico
(SAPW_Materia_Nivel)",(SELECT Current_TimeStamp));

END IF;

```

        END IF;
        UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
(SELECT Current_Date);
        RETURN V_PeAc_Materia;
    END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Plana Docente

```

DROP FUNCTION
SAPW_Mant_PlanaDocente(INTEGER,VARCHAR,INTEGER,VARCHAR);
CREATE OR REPLACE FUNCTION
SAPW_Mant_PlanaDocente(INTEGER,VARCHAR,INTEGER,VARCHAR)
RETURNS VARCHAR AS '
    DECLARE
        V_Cab_Auditoria          ALIAS FOR $1;
        V_PIDo_Docente           ALIAS FOR $2;
        V_PIDo_Carrera           ALIAS FOR $3;
        V_Tipo                   ALIAS FOR $4;
    BEGIN
        IF V_Tipo = "I" THEN
            INSERT INTO SAPW_Docente_Carrera
VALUES(V_PIDo_Docente, V_PIDo_Carrera);
            INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Plana Docente se agrego al Docente con Cédula " || V_PIDo_Docente || "
Perteneiente a la Carrera Universitaria con Código " || V_PIDo_Carrera::TEXT,
"Se trabajo en la tabla de Plana Docente (SAPW_Docente_Carrera)",(SELECT
Current_TimeStamp));
        ELSE
            IF V_Tipo = "D" THEN
                DELETE FROM SAPW_Docente_Carrera WHERE
Carr_Codigo = V_PIDo_Carrera AND Pers_Cedula = V_PIDo_Docente;
                INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Plana Docente se Eliminó al Docente con Cédula " || V_PIDo_Docente || "
Perteneiente a la Carrera Universitaria con Código " || V_PIDo_Carrera::TEXT,
"Se trabajo en la tabla de Plana Docente
(SAPW_Docente_Carrera)",(SELECT Current_TimeStamp));
            END IF;
        END IF;
        UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
(SELECT Current_Date);
        RETURN V_PIDo_Docente;
    END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Periodo Lectivo

DROP FUNCTION(INTEGER,INTEGER,INTEGER,VARCHAR);

SAPW_Mant_PeriodoLectivo(INTEGER,INTEGER,INTEGER,VARCHAR);

CREATE FUNCTION

SAPW_Mant_PeriodoLectivo(INTEGER,INTEGER,INTEGER,VARCHAR)

RETURNS VARCHAR AS '

DECLARE

V_Cab_Auditoria ALIAS FOR \$1;

V_PeLe_Anio ALIAS FOR \$2;

V_PeLe_Carrera ALIAS FOR \$3;

V_Tipo ALIAS FOR \$4;

BEGIN

IF V_Tipo = "I" THEN

INSERT INTO SAPW_Anio_Carrera VALUES(V_PeLe_Anio,
V_PeLe_Carrera,NEXTVAL("SAPW_S_Foto_Cabecera"),2,7);

INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Periodo Lectivo se agrego la Carrera Universitaria con Código " ||
V_PeLe_Carrera::TEXT || " Perteneciente al Año con Código " || V_PeLe_Anio::TEXT,
"Se trabajo en la tabla de Periodo Lectivo (SAPW_Anio_Carrera)",(SELECT
Current_TimeStamp));

ELSE

IF V_Tipo = "D" THEN

DELETE FROM SAPW_Anio_Carrera WHERE
Carr_Codigo = V_PeLe_Carrera AND Anio_Codigo = V_PeLe_Anio;

INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Periodo Lectivo se Eliminó la Carrera Universitaria con Código " ||
V_PeLe_Carrera::TEXT || " Perteneciente al Año con Código " || V_PeLe_Anio::TEXT,
"Se trabajo en la tabla de Periodo Lectivo (SAPW_Anio_Carrera)",(SELECT
Current_TimeStamp));

END IF;

END IF;

RETURN V_PeLe_Anio;

END;

' LANGUAGE 'plpgsql';

--Función para el Mantenimiento de Periodos de Carreras Universitarias

DROP FUNCTION

SAPW_Mant_PeriodoCarrera(INTEGER,INTEGER,INTEGER,INTEGER,VARCHAR
,VARCHAR);

CREATE OR REPLACE FUNCTION

SAPW_Mant_PeriodoCarrera(INTEGER,INTEGER,INTEGER,INTEGER,VARCHAR
,VARCHAR) RETURNS INTEGER AS '

DECLARE

V_Cab_Auditoria ALIAS FOR \$1;

V_PeCa_Anio ALIAS FOR \$2;

V_PeCa_Carrera ALIAS FOR \$3;

V_PeCa_Codigo ALIAS FOR \$4;

V_PeCa_Descripcion ALIAS FOR \$5;

```

V_Tipo                ALIAS FOR $6;
V_Secuencia           INTEGER;

BEGIN
    IF V_Tipo = "I" THEN
        SELECT NEXTVAL ("SAPW_S_Periodo") INTO V_Secuencia;
        INSERT INTO SAPW_Periodo_Anio_Carrera
VALUES(V_PeCa_Anio, V_PeCa_Carrera, V_Secuencia, V_PeCa_Descripcion, 2, 7);
        INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Periodos de Carreras Universitarias se Creo el registro con Código " ||
V_Secuencia::TEXT || " Perteneciente a la Carrera Universitaria con Código " ||
V_PeCa_Carrera::TEXT,
                                "Se trabajo en la tabla de Periodos de Carreras
Universitarias (SAPW_Periodo_Anio_Carrera)",(SELECT Current_TimeStamp));
    ELSE
        V_Secuencia := V_PeCa_Codigo;
        IF V_Tipo = "U" THEN
            UPDATE SAPW_Periodo_Anio_Carrera SET
Peri_Descripcion = V_PeCa_Descripcion
                                WHERE Carr_Codigo =
V_PeCa_Carrera AND Peri_Codigo = V_PeCa_Codigo AND
                                Anio_Codigo =
V_PeCa_Anio;
            INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Periodos de Carreras Universitarias se Modificó el registro con Código " ||
V_PeCa_Codigo::TEXT || " Perteneciente a la Carrera Universitaria con Código " ||
V_PeCa_Carrera::TEXT, "Se trabajo en la tabla de Periodos de Carreras Universitarias
(SAPW_Periodo_Anio_Carrera)",(SELECT Current_TimeStamp));
        ELSE
            IF V_Tipo = "D" THEN
                DELETE FROM SAPW_Periodo_Anio_Carrera
WHERE Carr_Codigo = V_PeCa_Carrera AND Peri_Codigo = V_PeCa_Codigo AND
Anio_Codigo = V_PeCa_Anio;
                INSERT INTO SAPW_Detalle_Auditoria
VALUES (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
Mantenimiento de Periodos de Carreras Universitarias se Eliminó el registro con
Código " || V_PeCa_Codigo::TEXT || " Perteneciente a la Carrera Universitaria con
Código " || V_PeCa_Carrera::TEXT,
                                "Se trabajo en la tabla de Periodos
de Carreras Universitarias (SAPW_Periodo_Anio_Carrera)",(SELECT
Current_TimeStamp));
            END IF;
        END IF;
    END IF;
    RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

```


--Función para el Mantenimiento de Paralelos

DROP FUNCTION

SAPW_Mant_Paralelos(INTEGER,INTEGER,INTEGER,INTEGER,INTEGER,INTEGER,INTEGER,VARCHAR,VARCHAR,VARCHAR);

CREATE OR REPLACE FUNCTION

SAPW_Mant_Paralelos(INTEGER,INTEGER,INTEGER,INTEGER,INTEGER,INTEGER,INTEGER,VARCHAR,VARCHAR,VARCHAR) RETURNS INTEGER AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_Para_Anio	ALIAS FOR \$2;
V_Para_Periodo	ALIAS FOR \$3;
V_Para_Paralelo	ALIAS FOR \$4;
V_Para_Nivel	ALIAS FOR \$5;
V_Para_Carrera	ALIAS FOR \$6;
V_Para_Materia	ALIAS FOR \$7;
V_Para_Docente	ALIAS FOR \$8;
V_Tipo	ALIAS FOR \$9;
V_Secuencia	INTEGER;

BEGIN

IF V_Tipo = "I" THEN

SELECT NEXTVAL ("SAPW_S_Paralelos") INTO

V_Secuencia;

INSERT INTO SAPW_Paralelo VALUES(V_Para_Anio,
V_Para_Periodo, V_Secuencia, V_Para_Nivel, V_Para_Carrera, V_Para_Materia,
V_Para_Docente);

INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Paralelos se Creo el registro con Código " || V_Secuencia::TEXT || " de la Materia
con Código " || V_Para_Materia || " del Nivel " || V_Para_Nivel::TEXT || " en el Año " ||
V_Para_Anio::TEXT || " y Termino " || V_Para_Periodo::TEXT || " Perteneciente a la
Carrera Universitaria con Código " || V_Para_Carrera::TEXT,
"Se trabajo en la tabla de Paralelos
(SAPW_Paralelo)",(SELECT Current_TimeStamp));

ELSE

V_Secuencia := V_Para_Paralelo;

IF V_Tipo = "D" THEN

DELETE FROM SAPW_Paralelo WHERE Carr_Codigo
= V_Para_Carrera AND Peri_Codigo = V_Para_Periodo AND
Anio_Codigo =
V_Para_Anio AND Mate_Codigo = V_Para_Materia AND
NiAc_Codigo =
V_Para_Nivel AND Para_Paralelo = V_Para_Paralelo;

INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Paralelos se Eliminó el registro con Código " || V_Secuencia::TEXT || " de la Materia
con Código " || V_Para_Materia || " del Nivel " || V_Para_Nivel::TEXT || " en el Año " ||
V_Para_Anio::TEXT || " y Termino " || V_Para_Periodo::TEXT || " Perteneciente a la
Carrera Universitaria con Código " || V_Para_Carrera::TEXT,

"Se trabajo en la tabla de Paralelos
(SAPW_Paralelo)",(SELECT Current_TimeStamp));


```

        END IF;
    END IF;
    UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
(SELECT Current_Date);
    RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Distribución de Paralelos

DROP FUNCTION

SAPW_Mant_DistribucionParalelos(INTEGER,INTEGER,INTEGER,INTEGER,VAR
CHAR,INTEGER,INTEGER,INTEGER,INTEGER, INTEGER, INTEGER, INTEGER,
VARCHAR);

CREATE OR REPLACE FUNCTION

SAPW_Mant_DistribucionParalelos(INTEGER,INTEGER,INTEGER,INTEGER,VAR
CHAR,INTEGER,INTEGER,INTEGER,INTEGER, INTEGER, INTEGER, INTEGER,
VARCHAR) RETURNS VOID AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_HoPa_Codigo	ALIAS FOR \$2;
V_HoPa_Nivel	ALIAS FOR \$3;
V_HoPa_Carrera	ALIAS FOR \$4;
V_HoPa_Materia	ALIAS FOR \$5;
V_HoPa_Paralelo	ALIAS FOR \$6;
V_HoPa_Anio	ALIAS FOR \$7;
V_HoPa_Periodo	ALIAS FOR \$8;
V_HoPa_Dia	ALIAS FOR \$9;
V_HoPa_Aula	ALIAS FOR \$10;
V_HoPa_Tipo	ALIAS FOR \$11;
V_Tipo	ALIAS FOR \$12;

BEGIN

IF V_Tipo = "I" THEN

INSERT INTO SAPW_Horario_Paralelo

VALUES(V_HoPa_Codigo, V_HoPa_Nivel, V_HoPa_Carrera, V_HoPa_Materia,
V_HoPa_Paralelo,

V_HoPa_Anio,

V_HoPa_Periodo, V_HoPa_Dia, V_HoPa_Aula, V_HoPa_Tipo);

INSERT INTO SAPW_Detalle_Auditoria VALUES

(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Distribución de Paralelos se Creo el Horario con Código " || V_HoPa_Codigo::TEXT
|| " del Paralelo " || V_HoPa_Paralelo::TEXT || " de la Materia con Código " ||
V_HoPa_Materia || " del Nivel " || V_HoPa_Nivel::TEXT || " en el Año " ||
V_HoPa_Anio::TEXT || " y Termino " || V_HoPa_Periodo::TEXT || " Perteneciente a la
Carrera Universitaria con Código " || V_HoPa_Carrera::TEXT,

"Se trabajo en la tabla de Distribución de Paralelos

(SAPW_Horario_Paralelo)",(SELECT Current_TimeStamp));

ELSE

IF V_Tipo = "D" THEN

```

DELETE FROM SAPW_Horario_Paralelo WHERE
Carr_Codigo = V_HoPa_Carrera AND Peri_Codigo = V_HoPa_Periodo AND
Anio_Codigo =
V_HoPa_Anio AND Mate_Codigo = V_HoPa_Materia AND
NiAc_Codigo =
V_HoPa_Nivel AND Para_Paralelo = V_HoPa_Paralelo AND
Hora_Codigo =
V_HoPa_Codigo AND HoPa_Dia = V_HoPa_Dia;
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Distribución de Paralelos se Eliminó el Horario con Código " ||
V_HoPa_Codigo::TEXT || " del Paralelo " || V_HoPa_Paralelo::TEXT || " de la
Materia con Código " || V_HoPa_Materia || " del Nivel " || V_HoPa_Nivel::TEXT || " en
el Año " || V_HoPa_Anio::TEXT || " y Termino " || V_HoPa_Periodo::TEXT || "
Pertenece a la Carrera Universitaria con Código " || V_HoPa_Carrera::TEXT,
"Se trabajo en la tabla de Distribución de
Paralelos (SAPW_Horario_Paralelo)",(SELECT Current_TimeStamp));
END IF;
END IF;
UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
(SELECT Current_Date);
RETURN 0;
END;
' LANGUAGE 'plpgsql';

```

```

--Función para el Mantenimiento de Estudiantes a Carreras Universitarias
DROP FUNCTION SAPW_Mant_EstudianteCarrera(INTEGER,VARCHAR,
INTEGER, VARCHAR);
CREATE OR REPLACE FUNCTION
SAPW_Mant_EstudianteCarrera(INTEGER,VARCHAR, INTEGER, VARCHAR)
RETURNS VOID AS '
DECLARE
    V_Cab_Auditoria          ALIAS FOR $1;
    V_EsCa_Estudiante        ALIAS FOR $2;
    V_EsCa_Carrera           ALIAS FOR $3;
    V_Tipo                   ALIAS FOR $4;
BEGIN
    IF V_Tipo = "I" THEN
        INSERT INTO SAPW_Estudiante_Carrera
VALUES(V_EsCa_Estudiante, V_EsCa_Carrera, 1);
        INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Estudiantes a Carreras Universitarias se Registro al Estudiante con Matricula " ||
V_EsCa_Estudiante || " a la Carrera Universitaria con Código " ||
V_EsCa_Carrera::TEXT,
"Se trabajo en la tabla de Estudiantes a Carreras
Universitarias (SAPW_Estudiante_Carrera)",(SELECT Current_TimeStamp));
    ELSE
        IF V_Tipo = "D" THEN

```

```

DELETE FROM SAPW_Estudiante_Carrera WHERE
Carr_Codigo = V_EsCa_Carrera AND Estu_Matricula = V_EsCa_Estudiante;
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Estudiantes a Carreras Universitarias se Eliminó al Estudiante con Matricula " ||
V_EsCa_Estudiante || " a la Carrera Universitaria con Código " ||
V_EsCa_Carrera::TEXT,
"Se trabajo en la tabla de Estudiantes a
Carreras Universitarias (SAPW_Estudiante_Carrera)",(SELECT Current_TimeStamp));
END IF;
END IF;
RETURN 0;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Inscripción de Estudiantes

DROP FUNCTION

SAPW_Mant_RegistroEstudiantes(INTEGER,VARCHAR,INTEGER,INTEGER,VAR
CHAR,INTEGER,INTEGER,INTEGER,INTEGER,INTEGER,VARCHAR);

CREATE OR REPLACE FUNCTION

SAPW_Mant_RegistroEstudiantes(INTEGER,VARCHAR,INTEGER,INTEGER,VAR
CHAR,INTEGER,INTEGER,INTEGER,INTEGER,INTEGER,VARCHAR) RETURNS VOID
AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_EsPa_Estudiante	ALIAS FOR \$2;
V_EsPa_Nivel	ALIAS FOR \$3;
V_EsPa_Carrera	ALIAS FOR \$4;
V_EsPa_Materia	ALIAS FOR \$5;
V_EsPa_Paralelo	ALIAS FOR \$6;
V_EsPa_Anio	ALIAS FOR \$7;
V_EsPa_Periodo	ALIAS FOR \$8;
V_EsPa_Estado	ALIAS FOR \$9;
V_Tipo	ALIAS FOR \$10;

BEGIN

IF V_Tipo = "I" THEN

INSERT INTO SAPW_Estudiante_Paralelo

VALUES(V_EsPa_Estudiante, V_EsPa_Nivel, V_EsPa_Carrera, V_EsPa_Materia,
V_EsPa_Paralelo,

V_EsPa_Anio,

V_EsPa_Periodo, 0, V_EsPa_Estado);

INSERT INTO SAPW_Detalle_Auditoria VALUES

(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Inscripción de Estudiantes se registro al Estudiante con Matricula " ||

V_EsPa_Estudiante || " al Paralelo " || V_EsPa_Paralelo::TEXT || " de la Materia con
Código " || V_EsPa_Materia || " del Nivel " || V_EsPa_Nivel::TEXT || " en el Año " ||

V_EsPa_Anio::TEXT || " y Termino " || V_EsPa_Periodo::TEXT || " Perteneciente a la
Carrera Universitaria con Código " || V_EsPa_Carrera::TEXT, "Se trabajo en la tabla de

```

Inscripción de Estudiantes (SAPW_Estudiante_Paralelo)",(SELECT
Current_TimeStamp));
    ELSE
        IF V_Tipo = "U" THEN
            UPDATE SAPW_Estudiante_Paralelo SET Esta_Codigo = V_EsPa_Estado WHERE
            Carr_Codigo = V_EsPa_Carrera AND Peri_Codigo = V_EsPa_Periodo AND
            Anio_Codigo = V_EsPa_Anio AND Mate_Codigo = V_EsPa_Materia AND
            NiAc_Codigo = V_EsPa_Nivel AND Para_Paralelo = V_EsPa_Paralelo AND
            Estu_Matricula = V_EsPa_Estudiante;
            INSERT INTO SAPW_Detalle_Auditoria VALUES
            (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
            de Inscripción de Estudiantes se Modificó al Estudiante con Matricula " ||
            V_EsPa_Estudiante || " al Paralelo " || V_EsPa_Paralelo::TEXT || " de la Materia con
            Código " || V_EsPa_Materia || " del Nivel " || V_EsPa_Nivel::TEXT || " en el Año " ||
            V_EsPa_Anio::TEXT || " y Termino " || V_EsPa_Periodo::TEXT || " Perteneciente a la
            Carrera Universitaria con Código " || V_EsPa_Carrera::TEXT, "Se trabajo en la tabla de
            Inscripción de Estudiantes (SAPW_Estudiante_Paralelo)",(SELECT
            Current_TimeStamp));
        ELSE
            IF V_Tipo = "D" THEN
                DELETE FROM SAPW_Estudiante_Paralelo WHERE Carr_Codigo =
                V_EsPa_Carrera AND Peri_Codigo = V_EsPa_Periodo AND
                Anio_Codigo = V_EsPa_Anio AND Mate_Codigo = V_EsPa_Materia AND
                NiAc_Codigo = V_EsPa_Nivel AND Para_Paralelo = V_EsPa_Paralelo AND
                Estu_Matricula = V_EsPa_Estudiante;
                INSERT INTO SAPW_Detalle_Auditoria VALUES
                (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
                de Inscripción de Estudiantes se Eliminó al Estudiante con Matricula " ||
                V_EsPa_Estudiante || " al Paralelo " || V_EsPa_Paralelo::TEXT || " de la Materia con
                Código " || V_EsPa_Materia || " del Nivel " || V_EsPa_Nivel::TEXT || " en el Año " ||
                V_EsPa_Anio::TEXT || " y Termino " || V_EsPa_Periodo::TEXT || " Perteneciente a la
                Carrera Universitaria con Código " || V_EsPa_Carrera::TEXT, "Se trabajo en la tabla de
                Inscripción de Estudiantes (SAPW_Estudiante_Paralelo)",(SELECT
                Current_TimeStamp));
            END IF;
        END IF;
    END IF;
    UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
    (SELECT Current_Date);
    RETURN 0;
END;
' LANGUAGE 'plpgsql';

```



--Función para el Mantenimiento de Deudas Estudiantiles

```
DROP FUNCTION SAPW_Mant_DeudaEstudiante(INTEGER,
INTEGER,VARCHAR,INTEGER,VARCHAR,DATE,FLOAT,VARCHAR);
CREATE OR REPLACE FUNCTION
SAPW_Mant_DeudaEstudiante(INTEGER,INTEGER,VARCHAR,INTEGER,VARCH
AR,DATE,FLOAT,VARCHAR) RETURNS VOID AS '
```

```
DECLARE
```

```
    V_Cab_Auditoria          ALIAS FOR $1;
    V_DeEs_Codigo            ALIAS FOR $2;
    V_DeEs_Estudiante        ALIAS FOR $3;
    V_DeEs_Carrera           ALIAS FOR $4;
    V_DeEs_Motivo            ALIAS FOR $5;
    V_DeEs_Fecha             ALIAS FOR $6;
    V_DeEs_Valor             ALIAS FOR $7;
    V_Tipo                   ALIAS FOR $8;
    V_Secuencia              INTEGER;
```

```
BEGIN
```

```
    IF V_Tipo = "I" THEN
```

```
        SELECT NEXTVAL("SAPW_S_Deuda_Estudiante") INTO
```

```
V_Secuencia;
```

```
        INSERT INTO SAPW_Deuda_Estudiante
```

```
VALUES(V_Secuencia, V_DeEs_Estudiante, V_DeEs_Carrera, V_DeEs_Motivo,
V_DeEs_Fecha, V_DeEs_Valor);
```

```
        INSERT INTO SAPW_Detalle_Auditoria VALUES
```

```
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Deudas Estudiantiles se registro una deuda con Código " || V_Secuencia::TEXT || " al
Estudiante con Matricula " || V_DeEs_Estudiante || " de la Carrera Universitaria con
Código " || V_DeEs_Carrera::TEXT, "Se trabajo en la tabla de Deudas Estudiantiles
(SAPW_Deuda_Estudiante)",(SELECT Current_TimeStamp));
```

```
    ELSE
```

```
        IF V_Tipo = "U" THEN
```

```
            UPDATE SAPW_Deuda_Estudiante SET DeEs_Motivo =
```

```
V_DeEs_Motivo, DeEs_Valor = V_DeEs_Valor, DeEs_Fecha_Vencimiento =
V_DeEs_Fecha WHERE Carr_Codigo = V_DeEs_Carrera AND Estu_Matricula =
V_DeEs_Estudiante AND DeEs_Linea = V_DeEs_Codigo;
```

```
            INSERT INTO SAPW_Detalle_Auditoria VALUES
```

```
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Deudas Estudiantiles se Modificó una deuda con Código " || V_DeEs_Codigo::TEXT
|| " al Estudiante con Matricula " || V_DeEs_Estudiante || " de la Carrera Universitaria
con Código " || V_DeEs_Carrera::TEXT, "Se trabajo en la tabla de Deudas Estudiantiles
(SAPW_Deuda_Estudiante)",(SELECT Current_TimeStamp));
```

```
        ELSE
```

```
            IF V_Tipo = "D" THEN
```

```
DELETE FROM SAPW_Deuda_Estudiante WHERE Carr_Codigo = V_DeEs_Carrera
AND Estu_Matricula = V_DeEs_Estudiante AND DeEs_Linea = V_DeEs_Codigo;
```

```
INSERT INTO SAPW_Detalle_Auditoria VALUES
```

```
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Deudas Estudiantiles se Eliminó una deuda con Código " || V_DeEs_Codigo::TEXT
|| " al Estudiante con Matricula " || V_DeEs_Estudiante || " de la Carrera Universitaria
```



```

con Código " || V_DeEs_Carrera::TEXT, "Se trabajo en la tabla de Deudas Estudiantiles
(SAPW_Deuda_Estudiante)",(SELECT Current_TimeStamp));
        END IF;
    END IF;
END IF;
UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
(SELECT Current_Date);
RETURN 0;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Promociones Graduadas

DROP FUNCTION

SAPW_Mant_PromocionesGraduadas(INTEGER,INTEGER,INTEGER,INTEGER,VA
RCHAR,VARCHAR,VARCHAR,FLOAT,VARCHAR,VARCHAR);

CREATE OR REPLACE FUNCTION

SAPW_Mant_PromocionesGraduadas(INTEGER,INTEGER,INTEGER,INTEGER,VA
RCHAR,VARCHAR,VARCHAR,FLOAT,VARCHAR,VARCHAR) RETURNS VOID
AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_Prom_Codigo	ALIAS FOR \$2;
V_Prom_Anio	ALIAS FOR \$3;
V_Prom_Carrera	ALIAS FOR \$4;
V_Prom_Estudiante	ALIAS FOR \$5;
V_Prom_Nombre	ALIAS FOR \$6;
V_Prom_Mail	ALIAS FOR \$7;
V_Prom_Promedio	ALIAS FOR \$8;
V_Prom_SitioWebUrl	ALIAS FOR \$9;
V_Tipo	ALIAS FOR \$10;
V_Secuencia	INTEGER;

BEGIN

IF V_Tipo = "I" THEN

SELECT NEXTVAL("SAPW_S_Promocion") INTO

V_Secuencia;

INSERT INTO SAPW_Promocion VALUES(V_Secuencia,
V_Prom_Anio, V_Prom_Carrera, V_Prom_Estudiante, V_Prom_Nombre,
V_Prom_Mail, V_Prom_Promedio, V_Prom_SitioWebUrl);

INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Promociones Graduadas se creo el registro con Código " || V_Secuencia::TEXT || " a
la Carrera Universitaria con Código " || V_Prom_Carrera::TEXT, "Se trabajo en la tabla
de Promociones Graduadas (SAPW_Promocion)",(SELECT Current_TimeStamp));

ELSE

IF V_Tipo = "U" THEN

UPDATE SAPW_Promocion SET Estu_Matricula =
V_Prom_Estudiante, Prom_Nombre = V_Prom_Nombre, Prom_Mail = V_Prom_Mail,
Prom_Promedio = V_Prom_Promedio, Prom_SitioWebUrl = V_Prom_SitioWebUrl
WHERE Prom_Codigo = V_Prom_Codigo;


```

INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Promociones Graduadas se Modificó el registro con Código " ||
V_Prom_Codigo::TEXT || " a la Carrera Universitaria con Código " ||
V_Prom_Carrera::TEXT, "Se trabajo en la tabla de Promociones Graduadas
(SAPW_Promocion)",(SELECT Current_TimeStamp));
ELSE
IF V_Tipo = "D" THEN
DELETE FROM SAPW_Promocion WHERE Prom_Codigo = V_Prom_Codigo;
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Promociones Graduadas se Eliminó el registro con Código " ||
V_Prom_Codigo::TEXT || " a la Carrera Universitaria con Código " ||
V_Prom_Carrera::TEXT, "Se trabajo en la tabla de Promociones Graduadas
(SAPW_Promocion)",(SELECT Current_TimeStamp));
END IF;
END IF;
END IF;
UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
(SELECT Current_Date);
RETURN 0;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Fotos

DROP FUNCTION

SAPW_Mant_Fotos(INTEGER,INTEGER,INTEGER,VARCHAR,VARCHAR,VARCHAR,INTEGER,VARCHAR);

CREATE OR REPLACE FUNCTION

SAPW_Mant_Fotos(INTEGER,INTEGER,INTEGER,VARCHAR,VARCHAR,VARCHAR,INTEGER,VARCHAR) RETURNS VOID AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_Foto_Codigo	ALIAS FOR \$2;
V_Foto_Numero	ALIAS FOR \$3;
V_Foto_Direccion	ALIAS FOR \$4;
V_Foto_Thumbnails	ALIAS FOR \$5;
V_Foto_Descripción	ALIAS FOR \$6;
V_Foto_Estado	ALIAS FOR \$7;
V_Tipo	ALIAS FOR \$8;

BEGIN

IF V_Tipo = "I" THEN

INSERT INTO SAPW_Foto VALUES(V_Foto_Codigo,
V_Foto_Numero, V_Foto_Direccion, V_Foto_Thumbnails, NULL,
V_Foto_Descripcion, V_Foto_Estado);

ELSE

IF V_Tipo = "U" THEN

```

UPDATE SAPW_Foto SET Foto_Direccion =
V_Foto_Direccion, Foto_Estado = V_Foto_Estado, Foto_Thumbnails =
V_Foto_Thumbnails, Foto_Descripcion = V_Foto_Descripcion
WHERE Foto_Codigo = V_Foto_Codigo AND Foto_Numero = V_Foto_Numero;
ELSE
IF V_Tipo = "D" THEN
DELETE FROM SAPW_Foto WHERE Foto_Codigo = V_Foto_Codigo AND
Foto_Numero = V_Foto_Numero;
END IF;
END IF;
END IF;
RETURN 0;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Forums Paralelos

DROP FUNCTION

SAPW_Mant_ForumClase(INTEGER,INTEGER,INTEGER,INTEGER,VARCHAR,IN
TEGER,INTEGER,INTEGER,VARCHAR,VARCHAR,DATE,VARCHAR);

CREATE OR REPLACE FUNCTION

SAPW_Mant_ForumClase(INTEGER,INTEGER,INTEGER,INTEGER,VARCHAR,IN
TEGER,INTEGER,INTEGER,VARCHAR,VARCHAR,DATE,VARCHAR)

RETURNS INTEGER AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_FoCl_Numero	ALIAS FOR \$2;
V_FoCl_Nivel	ALIAS FOR \$3;
V_FoCl_Carrera	ALIAS FOR \$4;
V_FoCl_Materia	ALIAS FOR \$5;
V_FoCl_Paralelo	ALIAS FOR \$6;
V_FoCl_Anio	ALIAS FOR \$7;
V_FoCl_Periodo	ALIAS FOR \$8;
V_FoCl_Nombre	ALIAS FOR \$9;
V_FoCl_Descripción	ALIAS FOR \$10;
V_FoCl_Fecha	ALIAS FOR \$11;
V_Tipo	ALIAS FOR \$12;
V_Secuencia	INTEGER;

BEGIN

IF V_Tipo = "I" THEN

SELECT NEXTVAL("SAPW_S_Forum_Clase") INTO

V_Secuencia;

INSERT INTO SAPW_Forum_Clase VALUES(V_Secuencia,
V_FoCl_Nivel, V_FoCl_Carrera, V_FoCl_Materia, V_FoCl_Paralelo, V_FoCl_Anio,
V_FoCl_Periodo, V_FoCl_Nombre, V_FoCl_Descripcion, V_FoCl_Fecha);

INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Forums Paralelo se creo un registro con Código " || V_Secuencia::TEXT || "
Pertenece al Paralelo " || V_FoCl_Paralelo::TEXT || " de la Materia con Código " ||
V_FoCl_Materia || " del Nivel " || V_FoCl_Nivel::TEXT || " en el Año " ||


```

V_CaEs_Paralelo      ALIAS FOR $3;
V_CaEs_Anio          ALIAS FOR $4;
V_CaEs_Periodo       ALIAS FOR $5;
V_CaEs_Nivel         ALIAS FOR $6;
V_CaEs_Carrera       ALIAS FOR $7;
V_CaEs_Materia       ALIAS FOR $8;
V_CaEs_Titulo        ALIAS FOR $9;
V_CaEs_Descripcion   ALIAS FOR $10;
V_CaEs_NombreArchivo ALIAS FOR $11;
V_Tipo               ALIAS FOR $12;
V_Secuencia          INTEGER;

BEGIN
    IF V_Tipo = "I" THEN
        SELECT NEXTVAL("SAPW_S_Capitulos_Estudio") INTO
V_Secuencia;

        INSERT INTO SAPW_Capitulo_Estudio
VALUES(V_Secuencia, V_CaEs_Paralelo, V_CaEs_Anio, V_CaEs_Periodo,
V_CaEs_Nivel, V_CaEs_Carrera,
V_CaEs_Materia,
V_CaEs_Titulo, V_CaEs_Descripcion, V_CaEs_NombreArchivo);
        INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Capítulos de Estudio se creo el resgistro con Código " || V_Secuencia::TEXT || " en
el Paralelo " || V_CaEs_Paralelo::TEXT || " de la Materia con Código " ||
V_CaEs_Materia || " del Nivel " || V_CaEs_Nivel::TEXT || " en el Año " ||
V_CaEs_Anio::TEXT || " y Termino " || V_CaEs_Periodo::TEXT || " Perteneciente a la
Carrera Universitaria con Código " || V_CaEs_Carrera::TEXT, "Se trabajo en la tabla
de Capítulos de Estudio (SAPW_Capitulo_Estudio)",(SELECT Current_TimeStamp));
    ELSE
        IF V_Tipo = "U" THEN
            UPDATE SAPW_Capitulo_Estudio SET CaEs_Titulo =
V_CaEs_Titulo, CaEs_Descripcion = V_CaEs_Descripcion, CaEs_NombreArchivo =
V_CaEs_NombreArchivo WHERE Carr_Codigo = V_CaEs_Carrera AND Peri_Codigo
= V_CaEs_Periodo AND Anio_Codigo = V_CaEs_Anio AND Mate_Codigo =
V_CaEs_Materia AND NiAc_Codigo = V_CaEs_Nivel AND Para_Paralelo =
V_CaEs_Paralelo AND CaEs_Numero = V_CaEs_Numero;
            INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Capítulos de Estudio se Modificó el resgistro con Código " ||
V_CaEs_Numero::TEXT || " en el Paralelo " || V_CaEs_Paralelo::TEXT || " de la
Materia con Código " || V_CaEs_Materia || " del Nivel " || V_CaEs_Nivel::TEXT || " en
el Año " || V_CaEs_Anio::TEXT || " y Termino " || V_CaEs_Periodo::TEXT || "
Perteneciente a la Carrera Universitaria con Código " || V_CaEs_Carrera::TEXT,
"Se trabajo en la tabla de Capítulos de Estudio (SAPW_Capitulo_Estudio)",(SELECT
Current_TimeStamp));
        ELSE
            IF V_Tipo = "D" THEN
DELETE FROM SAPW_Capitulo_Estudio WHERE Carr_Codigo = V_CaEs_Carrera
AND Peri_Codigo = V_CaEs_Periodo AND Anio_Codigo = V_CaEs_Anio AND

```

```

Mate_Codigo = V_CaEs_Materia AND NiAc_Codigo = V_CaEs_Nivel AND
Para_Paralelo = V_CaEs_Paralelo AND CaEs_Numero = V_CaEs_Numero;
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Capítulos de Estudio se Eliminó el registro con Código " || V_CaEs_Numero::TEXT
|| " en el Paralelo " || V_CaEs_Paralelo::TEXT || " de la Materia con Código " ||
V_CaEs_Materia || " del Nivel " || V_CaEs_Nivel::TEXT || " en el Año " ||
V_CaEs_Anio::TEXT || " y Termino " || V_CaEs_Periodo::TEXT || " Pertenece a la
Carrera Universitaria con Código " || V_CaEs_Carrera::TEXT, "Se trabajo en la tabla
de Capítulos de Estudio (SAPW_Capitulo_Estudio)",(SELECT Current_TimeStamp));
END IF;
END IF;
END IF;
UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion = (SELECT
Current_Date);
RETURN 0;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Politicas del Curso

DROP FUNCTION

SAPW_Mant_PolíticasCurso(INTEGER,INTEGER,INTEGER,INTEGER,VARCHAR,
INTEGER,INTEGER,INTEGER,VARCHAR,VARCHAR);

CREATE OR REPLACE FUNCTION

SAPW_Mant_PolíticasCurso(INTEGER,INTEGER,INTEGER,INTEGER,VARCHAR,
INTEGER,INTEGER,INTEGER,VARCHAR,VARCHAR) RETURNS VOID AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_Poli_Numero	ALIAS FOR \$2;
V_Poli_Nivel	ALIAS FOR \$3;
V_Poli_Carrera	ALIAS FOR \$4;
V_Poli_Materia	ALIAS FOR \$5;
V_Poli_Paralelo	ALIAS FOR \$6;
V_Poli_Anio	ALIAS FOR \$7;
V_Poli_Periodo	ALIAS FOR \$8;
V_Poli_Descripcion	ALIAS FOR \$9;
V_Tipo	ALIAS FOR \$10;
V_Secuencia	INTEGER;

BEGIN

IF V_Tipo = "I" THEN

SELECT NEXTVAL("SAPW_S_Políticas") INTO V_Secuencia;

INSERT INTO SAPW_Política VALUES(V_Secuencia,

V_Poli_Nivel, V_Poli_Carrera, V_Poli_Materia, V_Poli_Paralelo, V_Poli_Anio,
V_Poli_Periodo, V_Poli_Descripcion);

INSERT INTO SAPW_Detalle_Auditoria VALUES

(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Políticas del Curso se creo el registro con Código " || V_Secuencia::TEXT || " en el
Paralelo " || V_Poli_Paralelo::TEXT || " de la Materia con Código " || V_Poli_Materia
|| " del Nivel " || V_Poli_Nivel::TEXT || " en el Año " || V_Poli_Anio::TEXT || " y


```

Termino " || V_Poli_Periodo::TEXT || " Perteneiente a la Carrera Universitaria con
Código " || V_Poli_Carrera::TEXT, "Se trabajo en la tabla de Politicas del Curso
(SAPW_Politica)",(SELECT Current_TimeStamp));
    ELSE
        IF V_Tipo = "U" THEN
            UPDATE SAPW_Politica SET Poli_Descripcion = V_Poli_Descripcion WHERE
            Carr_Codigo = V_Poli_Carrera AND Peri_Codigo = V_Poli_Periodo AND
            Anio_Codigo = V_Poli_Anio AND Mate_Codigo = V_Poli_Materia AND
            NiAc_Codigo = V_Poli_Nivel AND Para_Paralelo = V_Poli_Paralelo AND
            Poli_Numero = V_Poli_Numero;
            INSERT INTO SAPW_Detalle_Auditoria VALUES
            (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
            de Politicas del Curso se Modificó el resgistro con Código " || V_Poli_Numero::TEXT ||
            " en el Paralelo " || V_Poli_Paralelo::TEXT || " de la Materia con Código " ||
            V_Poli_Materia || " del Nivel " || V_Poli_Nivel::TEXT || " en el Año " ||
            V_Poli_Anio::TEXT || " y Termino " || V_Poli_Periodo::TEXT || " Perteneiente a la
            Carrera Universitaria con Código " || V_Poli_Carrera::TEXT, "Se trabajo en la tabla de
            Politicas del Curso (SAPW_Politica)",(SELECT Current_TimeStamp));
        ELSE
            IF V_Tipo = "D" THEN
                DELETE FROM SAPW_Politica WHERE Carr_Codigo = V_Poli_Carrera AND
                Peri_Codigo = V_Poli_Periodo AND Anio_Codigo = V_Poli_Anio AND Mate_Codigo
                = V_Poli_Materia AND NiAc_Codigo = V_Poli_Nivel AND Para_Paralelo =
                V_Poli_Paralelo AND Poli_Numero = V_Poli_Numero;
                INSERT INTO SAPW_Detalle_Auditoria VALUES
                (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
                de Politicas del Curso se Eliminó el resgistro con Código " || V_Poli_Numero::TEXT ||
                " en el Paralelo " || V_Poli_Paralelo::TEXT || " de la Materia con Código " ||
                V_Poli_Materia || " del Nivel " || V_Poli_Nivel::TEXT || " en el Año " ||
                V_Poli_Anio::TEXT || " y Termino " || V_Poli_Periodo::TEXT || " Perteneiente a la
                Carrera Universitaria con Código " || V_Poli_Carrera::TEXT,
                "Se trabajo en la tabla de Politicas del Curso (SAPW_Politica)",(SELECT
                Current_TimeStamp));
            END IF;
        END IF;
    END IF;
    UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion = (SELECT
    Current_Date);
    RETURN 0;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Deberes

DROP FUNCTION

SAPW_Mant_Deberes(INTEGER,INTEGER,INTEGER,INTEGER,INTEGER,INTEG
ER,INTEGER,INTEGER,VARCHAR,VARCHAR,DATE,DATE,VARCHAR,VARCH
AR);

CREATE OR REPLACE FUNCTION

SAPW_Mant_Deberes(INTEGER,INTEGER,INTEGER,INTEGER,INTEGER,INTEG

ER,INTEGER,INTEGER,VARCHAR,VARCHAR,DATE,DATE,VARCHAR,VARCHAR) RETURNS VOID AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_Debe_Numero	ALIAS FOR \$2;
V_Debe_Capitulo	ALIAS FOR \$3;
V_Debe_Paralelo	ALIAS FOR \$4;
V_Debe_Anio	ALIAS FOR \$5;
V_Debe_Periodo	ALIAS FOR \$6;
V_Debe_Nivel	ALIAS FOR \$7;
V_Debe_Carrera	ALIAS FOR \$8;
V_Debe_Materia	ALIAS FOR \$9;
V_Debe_Descripcion	ALIAS FOR \$10;
V_Debe_FechaPublicacion	ALIAS FOR \$11;
V_Debe_FechaEntrega	ALIAS FOR \$12;
V_Debe_NombreArchivo	ALIAS FOR \$13;
V_Tipo	ALIAS FOR \$14;
V_Secuencia	INTEGER;

BEGIN

IF V_Tipo = "T" THEN

SELECT NEXTVAL("SAPW_S_Deber") INTO V_Secuencia;

INSERT INTO SAPW_Deber VALUES(V_Secuencia,

V_Debe_Capitulo, V_Debe_Paralelo, V_Debe_Anio, V_Debe_Periodo,

V_Debe_Nivel, V_Debe_Carrera, V_Debe_Materia, V_Debe_Descripcion,

V_Debe_FechaPublicacion, V_Debe_FechaEntrega, V_Debe_NombreArchivo);

INSERT INTO SAPW_Detalle_Auditoria VALUES

(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento

de Deberes se creo el resgistro con Código " || V_Secuencia::TEXT || " del Capítulo " ||

V_Debe_Capitulo || " en el Paralelo " || V_Debe_Paralelo::TEXT || " de la Materia con

Código " || V_Debe_Materia || " del Nivel " || V_Debe_Nivel::TEXT || " en el Año " ||

V_Debe_Anio::TEXT || " y Termino " || V_Debe_Periodo::TEXT || " Perteneciente a la

Carrera Universitaria con Código " || V_Debe_Carrera::TEXT, "Se trabajo en la tabla de

Deberes (SAPW_Deber)",(SELECT Current_TimeStamp));

ELSE

IF V_Tipo = "U" THEN

UPDATE SAPW_Deber SET Debe_Descripcion =

V_Debe_Descripcion, Debe_FechaPublicacion = V_Debe_FechaPublicacion,

Debe_FechaEntrega = V_Debe_FechaEntrega, Debe_NombreArchivo =

V_Debe_NombreArchivo WHERE Carr_Codigo = V_Debe_Carrera AND Peri_Codigo =

= V_Debe_Periodo AND Anio_Codigo = V_Debe_Anio AND Mate_Codigo =

V_Debe_Materia AND NiAc_Codigo = V_Debe_Nivel AND Para_Paralelo =

V_Debe_Paralelo AND CaEs_Numero = V_Debe_Capitulo AND Debe_Numero =

V_Debe_Numero;

INSERT INTO SAPW_Detalle_Auditoria VALUES

(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento

de Deberes se Modificó el resgistro con Código " || V_Debe_Numero::TEXT || " del

Capítulo " || V_Debe_Capitulo || " en el Paralelo " || V_Debe_Paralelo::TEXT || " de la

Materia con Código " || V_Debe_Materia || " del Nivel " || V_Debe_Nivel::TEXT || " en

el Año " || V_Debe_Anio::TEXT || " y Termino " || V_Debe_Periodo::TEXT || "

Perteneciente a la Carrera Universitaria con Código " || V_Debe_Carrera::TEXT,

```

"Se trabajo en la tabla de Deberes (SAPW_Deber)",(SELECT Current_TimeStamp));
ELSE
    IF V_Tipo = "D" THEN
        DELETE FROM SAPW_Deber WHERE
        Carr_Codigo = V_Debe_Carrera AND Peri_Codigo = V_Debe_Periodo AND
        Anio_Codigo = V_Debe_Anio AND Mate_Codigo = V_Debe_Materia AND
        NiAc_Codigo = V_Debe_Nivel AND Para_Paralelo = V_Debe_Paralelo AND
        CaEs_Numero = V_Debe_Capitulo AND Debe_Numero = V_Debe_Numero;
        INSERT INTO SAPW_Detalle_Auditoria VALUES
        (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
        de Deberes se Modificó el resgistro con Código " || V_Debe_Numero::TEXT || " del
        Capítulo " || V_Debe_Capitulo || " en el Paralelo " || V_Debe_Paralelo::TEXT || " de la
        Materia con Código " || V_Debe_Materia || " del Nivel " || V_Debe_Nivel::TEXT || " en
        el Año " || V_Debe_Anio::TEXT || " y Termino " || V_Debe_Periodo::TEXT || "
        Perteneciente a la Carrera Universitaria con Código " || V_Debe_Carrera::TEXT,
        "Se trabajo en la tabla de Deberes (SAPW_Deber)",(SELECT Current_TimeStamp));
    END IF;
END IF;
UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
(SELECT Current_Date);
RETURN 0;
END;
' LANGUAGE 'plpgsql';

```



--Función para el Mantenimiento de Preguntas Estudiantes

DROP FUNCTION

SAPW_Mant_PreguntasEstudiante(INTEGER,INTEGER,INTEGER,VARCHAR,VAR
CHAR,VARCHAR,DATE,VARCHAR);

CREATE OR REPLACE FUNCTION

SAPW_Mant_PreguntasEstudiante(INTEGER,INTEGER,INTEGER,VARCHAR,VAR
CHAR,VARCHAR,DATE,VARCHAR) RETURNS VOID AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_PrEs_Numero	ALIAS FOR \$2;
V_PrEs_Forum	ALIAS FOR \$3;
V_PrEs_Pregunta	ALIAS FOR \$4;
V_PrEs_Persona	ALIAS FOR \$5;
V_PrEs_EMail	ALIAS FOR \$6;
V_PrEs_FechaEnvio	ALIAS FOR \$7;
V_Tipo	ALIAS FOR \$8;
V_Secuencia	INTEGER;

BEGIN

IF V_Tipo = "I" THEN

SELECT NEXTVAL("SAPW_S_Pregunta_Estudiante") INTO

V_Secuencia;

INSERT INTO SAPW_Pregunta_Estudiante

VALUES(V_Secuencia, V_PrEs_Forum, 1, V_PrEs_Pregunta, V_PrEs_Persona,
V_PrEs_EMail, V_PrEs_FechaEnvio);

```

INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Preguntas Forum Paralelo se creo el resgistro con Código " || V_Secuencia::TEXT || "
Perteneiente al Forum " || V_PrEs_Forum::TEXT,
"Se trabajo en la tabla de Pregunta Forum Paralelo
(SAPW_Pregunta_Estudiante)",(SELECT Current_TimeStamp));
ELSE
    IF V_Tipo = "U" THEN
        UPDATE SAPW_Pregunta_Estudiante SET
        PrEs_Pregunta = V_PrEs_Pregunta WHERE FoCl_Numero = V_PrEs_Forum AND
        PrEs_Numero = V_PrEs_Numero;
        INSERT INTO SAPW_Detalle_Auditoria VALUES
        (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
        de Preguntas Forum Paralelo se Modificó el resgistro con Código " ||
        V_PrEs_Numero::TEXT || " Perteneiente al Forum " || V_PrEs_Forum::TEXT,
        "Se trabajo en la tabla de Pregunta Forum Paralelo
        (SAPW_Pregunta_Estudiante)",(SELECT Current_TimeStamp));
    ELSE
        IF V_Tipo = "D" THEN
            DELETE FROM SAPW_Pregunta_Estudiante WHERE FoCl_Numero =
            V_PrEs_Forum AND PrEs_Numero = V_PrEs_Numero;
            INSERT INTO SAPW_Detalle_Auditoria VALUES
            (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
            de Preguntas Forum Paralelo se Eliminó el resgistro con Código " ||
            V_PrEs_Numero::TEXT || " Perteneiente al Forum " || V_PrEs_Forum::TEXT,
            "Se trabajo en la tabla de Pregunta Forum Paralelo
            (SAPW_Pregunta_Estudiante)",(SELECT Current_TimeStamp));
        END IF;
    END IF;
END IF;
RETURN 0;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Servicios por Departamento

DROP FUNCTION

SAPW_Mant_Servicios(INTEGER,INTEGER,INTEGER,VARCHAR,VARCHAR,VARCHAR,VARCHAR);

CREATE OR REPLACE FUNCTION

SAPW_Mant_Servicios(INTEGER,INTEGER,INTEGER,VARCHAR,VARCHAR,VARCHAR,VARCHAR) RETURNS VOID AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_Serv_Numero	ALIAS FOR \$2;
V_Serv_Departamento	ALIAS FOR \$3;
V_Serv_Nombre	ALIAS FOR \$4;
V_Serv_Descripcion	ALIAS FOR \$5;
V_Serv_Url	ALIAS FOR \$6;
V_Tipo	ALIAS FOR \$7;

```
V_Secuencia          INTEGER;

BEGIN
  IF V_Tipo = "I" THEN
    SELECT NEXTVAL ("SAPW_S_Servicios") INTO V_Secuencia;
    INSERT INTO SAPW_Servicio VALUES(V_Secuencia,
    V_Serv_Departamento, V_Serv_Nombre, V_Serv_Descripcion, 0,
    V_Serv_Url);
    INSERT INTO SAPW_Detalle_Auditoria VALUES
    (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
    de Servicios por Departamento se Creo el registro con Código " || V_Secuencia::TEXT ||
    " Perteneciente al Departamento con Código " || V_Serv_Departamento::TEXT,
    "Se trabajo en la tabla de Servicios por Departamento (SAPW_Servicio)",(SELECT
    Current_TimeStamp));
  ELSE
    IF V_Tipo = "U" THEN
      UPDATE SAPW_Servicio SET Serv_Nombre =
      V_Serv_Nombre, Serv_Descripcion = V_Serv_Descripcion, Serv_Url = V_Serv_Url
      WHERE Depa_Codigo = V_Serv_Departamento AND Serv_Numero =
      V_Serv_Numero;
      INSERT INTO SAPW_Detalle_Auditoria VALUES
      (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
      de Servicios por Departamento se Modificó el registro con Código " ||
      V_Serv_Numero::TEXT || " Perteneciente al Departamento con Código " ||
      V_Serv_Departamento::TEXT,
      "Se trabajo en la tabla de Servicios por Departamento (SAPW_Servicio)",(SELECT
      Current_TimeStamp));
    ELSE
      IF V_Tipo = "D" THEN
        DELETE FROM SAPW_Servicio WHERE Depa_Codigo = V_Serv_Departamento
        AND Serv_Numero = V_Serv_Numero;
        INSERT INTO SAPW_Detalle_Auditoria VALUES
        (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
        de Servicios por Departamento se Modificó el registro con Código " ||
        V_Serv_Numero::TEXT || " Perteneciente al Departamento con Código " ||
        V_Serv_Departamento::TEXT,
        "Se trabajo en la tabla de Servicios por Departamento (SAPW_Servicio)",(SELECT
        Current_TimeStamp));
      END IF;
    END IF;
  END IF;
  UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
  (SELECT Current_Date);
  RETURN 0;
END;
' LANGUAGE 'plpgsql';
```

--Función para el Mantenimiento de Productos

DROP FUNCTION

SAPW_Mant_Productos(INTEGER,INTEGER,INTEGER,INTEGER,INTEGER,VAR
CHAR,VARCHAR,FLOAT,VARCHAR,DATE,VARCHAR,VARCHAR);

CREATE OR REPLACE FUNCTION

SAPW_Mant_Productos(INTEGER,INTEGER,INTEGER,INTEGER,INTEGER,VAR
CHAR,VARCHAR,FLOAT,VARCHAR,DATE,VARCHAR,VARCHAR) RETURNS
VOID AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_Prod_Numero	ALIAS FOR \$2;
V_Prod_Servicio	ALIAS FOR \$3;
V_Prod_Departamento	ALIAS FOR \$4;
V_Prod_Cliente	ALIAS FOR \$5;
V_Prod_Nombre	ALIAS FOR \$6;
V_Prod_Descripción	ALIAS FOR \$7;
V_Prod_Costo	ALIAS FOR \$8;
V_Prod_Comentario	ALIAS FOR \$9;
V_Prod_Fecha	ALIAS FOR \$10;
V_Prod_Url	ALIAS FOR \$11;
V_Tipo	ALIAS FOR \$12;
V_Secuencia	INTEGER;

BEGIN

IF V_Tipo = "I" THEN

SELECT NEXTVAL ("SAPW_S_Productos") INTO

V_Secuencia;

INSERT INTO SAPW_Producto VALUES(V_Secuencia,
V_Prod_Servicio, V_Prod_Departamento, V_Prod_Cliente, V_Prod_Nombre,
V_Prod_Descripcion, V_Prod_Costo, V_Prod_Comentario, V_Prod_Fecha,
V_Prod_Url, 0, NEXTVAL("SAPW_S_Foto_Cabecera"));

INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Productos se Creo el registro con Código " || V_Secuencia::TEXT || " Perteneciente al
Servicio con código " || V_Prod_Servicio || " del Departamento con Código " ||
V_Prod_Departamento::TEXT, "Se trabajo en la tabla de Productos
(SAPW_Producto)",(SELECT Current_TimeStamp));

ELSE

IF V_Tipo = "U" THEN

UPDATE SAPW_Producto SET Prod_Nombre =
V_Prod_Nombre, Prod_Descripcion = V_Prod_Descripcion, Prod_Url = V_Prod_Url,
Clie_Codigo = V_Prod_Cliente, Prod_Costo = V_Prod_Costo, Prod_Comentario =
V_Prod_Comentario, Prod_Fecha = V_Prod_Fecha WHERE Depa_Codigo =
V_Prod_Departamento AND Serv_Numero = V_Prod_Servicio AND Prod_Numero =
V_Prod_Numero;

INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Productos se Modificó el registro con Código " || V_Prod_Numero::TEXT || "
Perteneciente al Servicio con código " || V_Prod_Servicio || " del Departamento con
Código " || V_Prod_Departamento::TEXT, "Se trabajo en la tabla de Productos
(SAPW_Producto)",(SELECT Current_TimeStamp));


```

ELSE
    IF V_Tipo = "D" THEN
        DELETE FROM SAPW_Producto WHERE Depa_Codigo = V_Prod_Departamento
        AND Serv_Numero = V_Prod_Servicio AND Prod_Numero = V_Prod_Numero;
        INSERT INTO SAPW_Detalle_Auditoria VALUES
        (NEXTVAL("SAPW_S_Detalle_Auditoria"), V_Cab_Auditoria, "En el Mantenimiento
        de Productos se Eliminó el registro con Código " || V_Prod_Numero::TEXT || "
        Perteneciente al Servicio con código " || V_Prod_Servicio || " del Departamento con
        Código " || V_Prod_Departamento::TEXT, "Se trabajo en la tabla de Productos
        (SAPW_Producto)", (SELECT Current_TimeStamp));
        END IF;
    END IF;
    UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
    (SELECT Current_Date);
    RETURN 0;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Certificados

```

DROP FUNCTION SAPW_Mant_Certificados(int4, int4, varchar, varchar, varchar,
varchar, varchar, varchar)

```

```

CREATE FUNCTION public.sapw_mant_certificados(int4, int4, varchar, varchar,
varchar, varchar, varchar, varchar) RETURNS int4 AS '

```

```

DECLARE

```

```

    V_Cab_Auditoria          ALIAS FOR $1;
    V_Cert_Codigo            ALIAS FOR $2;
    V_Cert_Nombre            ALIAS FOR $3;
    V_Cert_Dirigido          ALIAS FOR $4;
    V_Cert_Objetoivo         ALIAS FOR $5;
    V_Cert_Requerimiento     ALIAS FOR $6;
    V_Cert_Lugar             ALIAS FOR $7;
    V_Tipo                   ALIAS FOR $8;
    V_Secuencia              INTEGER;

```

```

BEGIN

```

```

    IF V_Tipo = "I" THEN
        SELECT NEXTVAL ("SAPW_S_Certificados") INTO V_Secuencia;
        INSERT INTO SAPW_Certificado VALUES(V_Secuencia,
        V_Cert_Nombre, V_Cert_Dirigido, V_Cert_Objetoivo,
        V_Cert_Requerimiento, V_Cert_Lugar);
        INSERT INTO SAPW_Detalle_Auditoria VALUES
        (NEXTVAL("SAPW_S_Detalle_Auditoria"), V_Cab_Auditoria, "En el
        Mantenimiento de Certificados se Creo el registro con Código " ||
        V_Secuencia::TEXT, "Se trabajo en la tabla de Certificados
        (SAPW_Certificado)", (SELECT Current_TimeStamp));
    ELSE

```

```

        V_Secuencia := V_Cert_Codigo;
        IF V_Tipo = "U" THEN

```



```

UPDATE SAPW_Certificado SET Cert_Nombre = V_Cert_Nombre, Cert_Objetivo =
V_Cert_Objetivo, Cert_Dirigido = V_Cert_Dirigido, Cert_Requerimiento =
V_Cert_Requerimiento, Cert_Lugar = V_Cert_Lugar WHERE Cert_Codigo =
V_Cert_Codigo;
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Certificados se Modificó el registro con Código " || V_Cert_Codigo::TEXT, "Se
trabajo en la tabla de Certificados (SAPW_Certificado)",(SELECT
Current_TimeStamp));
ELSE
    IF V_Tipo = "D" THEN
        DELETE FROM SAPW_Certificado WHERE
        Cert_Codigo = V_Cert_Codigo;
        INSERT INTO SAPW_Detalle_Auditoria
        VALUES(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento de
        Certificados se Eliminó el registro con Código " ||
        V_Cert_Codigo::TEXT, "Se trabajo en la tabla de
        Certificados (SAPW_Certificado)",(SELECT
        Current_TimeStamp));
    END IF;
END IF;
END IF;
UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
(SELECT Current_Date);
RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Seminarios

```

DROP FUNCTION SAPW_Mant_Seminarios(int4, int4, int4, varchar, varchar, int4,
float8, int4, varchar, varchar)
CREATE FUNCTION public.sapw_mant_seminarios(int4, int4, int4, varchar, varchar,
int4, float8, int4, varchar, varchar) RETURNS void AS '
DECLARE
    V_Cab_Auditoria          ALIAS FOR $1;
    V_Semi_Codigo            ALIAS FOR $2;
    V_Semi_Certificado        ALIAS FOR $3;
    V_Semi_Nombre            ALIAS FOR $4;
    V_Semi_Descripcion        ALIAS FOR $5;
    V_Semi_Duracion          ALIAS FOR $6;
    V_Semi_Costo             ALIAS FOR $7;
    V_Semi_Cupo             ALIAS FOR $8;
    V_Semi_Archivo           ALIAS FOR $9;
    V_Tipo                   ALIAS FOR $10;
    V_Secuencia              INTEGER;
BEGIN
    IF V_Tipo = "I" THEN

```

```

SELECT NEXTVAL ("SAPW_S_Seminarios") INTO
V_Secuencia;
INSERT INTO SAPW_Seminario VALUES(V_Secuencia,
V_Semi_Certificado, V_Semi_Nombre, V_Semi_Descripcion, V_Semi_Duracion,
V_Semi_Costo, V_Semi_Cupo, NULL, 12, V_Semi_Archivo);
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Certificados se Creo el Seminario con Código " || V_Secuencia::TEXT || "
Pertenece al Certificado " || V_Semi_Certificado::TEXT, "Se trabajo en la tabla de
Seminarios (SAPW_Seminario)",(SELECT Current_TimeStamp));
ELSE
IF V_Tipo = "U" THEN
UPDATE SAPW_Seminario SET Semi_Nombre =
V_Semi_Nombre, Semi_Descripcion = V_Semi_Descripcion, Semi_Duracion =
V_Semi_Duracion, Semi_Costo = V_Semi_Costo, Semi_Cupo = V_Semi_Cupo,
Semi_Archivo = V_Semi_Archivo WHERE Semi_Codigo = V_Semi_Codigo AND
Cert_Codigo = V_Semi_Certificado;
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Certificados se Modificó el Seminario con Código " || V_Semi_Codigo::TEXT || "
Pertenece al Certificado " || V_Semi_Certificado::TEXT, "Se trabajo en la tabla de
Seminarios (SAPW_Seminario)",(SELECT Current_TimeStamp));
ELSE
IF V_Tipo = "D" THEN
DELETE FROM SAPW_Seminario WHERE
Semi_Codigo = V_Semi_Codigo AND Cert_Codigo = V_Semi_Certificado;
INSERT INTO SAPW_Detalle_Auditoria
VALUES (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
Mantenimiento de Certificados se Eliminó el Seminario con Código " ||
V_Semi_Codigo::TEXT || " Pertenece al Certificado " ||
V_Semi_Certificado::TEXT, "Se trabajo en la tabla de Seminarios
(SAPW_Seminario)",(SELECT Current_TimeStamp));
END IF;
END IF;
END IF;
UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
(SELECT Current_Date);
RETURN 0;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Planificación de Certificados

```

DROP FUNCTION SAPW_Mod_FechaInicio_Planificacion(INTEGER,INTEGER,
INTEGER, DATE, INTEGER);
CREATE OR REPLACE FUNCTION
SAPW_Mod_FechaInicio_Planificacion(INTEGER,INTEGER, INTEGER, DATE,
INTEGER) RETURNS VOID AS '

```

```

DECLARE

```

```

V_Cab_Auditoria

```

```

ALIAS FOR $1;

```

```

V_Semi_Codigo      ALIAS FOR $2;
V_Semi_Certificado  ALIAS FOR $3;
V_Semi_FechaInicio ALIAS FOR $4;
V_Semi_Estado       ALIAS FOR $5;

BEGIN
    UPDATE SAPW_Seminario SET Semi_FechaInicio =
V_Semi_FechaInicio, Esta_Codigo = V_Semi_Estado
    WHERE Semi_Codigo = V_Semi_Codigo AND
Cert_Codigo = V_Semi_Certificado;
    INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Planificación de Certificados se Modificó la fecha de Inicio del Seminario con
Código " || V_Semi_Codigo::TEXT || " Perteneciente al Certificado " ||
V_Semi_Certificado::TEXT,
    "Se trabajo en la tabla de Seminarios
(SAPW_Seminario)","(SELECT Current_TimeStamp));
    UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
(SELECT Current_Date);
    RETURN 0;
END;
' LANGUAGE 'plpgsql';

```



```

--Función para el Mantenimiento de Planificación de Seminarios
DROP FUNCTION SAPW_Mant_HorarioSeminarios(INTEGER,INTEGER,
INTEGER, INTEGER, INTEGER, INTEGER, VARCHAR, VARCHAR, VARCHAR,
VARCHAR);
CREATE OR REPLACE FUNCTION
SAPW_Mant_HorarioSeminarios(INTEGER,INTEGER, INTEGER, INTEGER,
INTEGER, INTEGER, VARCHAR, VARCHAR, VARCHAR, VARCHAR)
RETURNS VOID AS '
    DECLARE
        V_Cab_Auditoria      ALIAS FOR $1;
        V_HoSe_Numero        ALIAS FOR $2;
        V_HoSe_Hora          ALIAS FOR $3;
        V_HoSe_Laboratorio    ALIAS FOR $4;
        V_HoSe_Certificado    ALIAS FOR $5;
        V_HoSe_Seminario      ALIAS FOR $6;
        V_HoSe_Profesor       ALIAS FOR $7;
        V_HoSe_Dias           ALIAS FOR $8;
        V_HoSe_Observacion    ALIAS FOR $9;
        V_Tipo                ALIAS FOR $10;
        V_Secuencia           INTEGER;
        V_Cupo               INTEGER;

    BEGIN
        IF V_Tipo = "I" THEN
            SELECT NEXTVAL ("SAPW_S_Horario_Seminario") INTO
V_Secuencia;
            SELECT Semi_Cupo INTO V_Cupo FROM SAPW_Seminario

```

```

WHERE Semi_Codigo =
V_HoSe_Seminario AND Cert_Codigo = V_HoSe_Certificado;
INSERT INTO SAPW_Horario_Seminario
VALUES(V_Secuencia, V_HoSe_Hora, V_HoSe_Laboratorio, V_HoSe_Certificado,
V_HoSe_Seminario, V_HoSe_Profesor, V_Cupo, V_HoSe_Dias,
V_HoSe_Observacion);
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Planificación de Certificados se Creo el Horario con Código " || V_Secuencia::TEXT
|| " Perteneiente al Seminario " || V_HoSe_Seminario::TEXT || " del Certificado " ||
V_HoSe_Certificado::TEXT, "Se trabajo en la tabla de Palnificación de Certificados
(SAPW_Horario_Seminario)",(SELECT Current_TimeStamp));
ELSE
IF V_Tipo = "U" THEN
UPDATE SAPW_Horario_Seminario SET Pers_Cedula =
V_HoSe_Profesor, HoSe_Dias = V_HoSe_Dias, Hora_Codigo = V_HoSe_Hora,
HoSe_Observacion = V_HoSe_Observacion, Labo_Codigo = V_HoSe_Laboratorio
WHERE HoSe_Numero = V_HoSe_Numero;
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Planificación de Certificados se Modificó el Horario con Código " ||
V_HoSe_Numero::TEXT || " Perteneiente al Seminario " || V_HoSe_Seminario::TEXT
|| " del Certificado " || V_HoSe_Certificado::TEXT, 'Se trabajo en la tabla de
Palnificación de Certificados (SAPW_Horario_Seminario)',(SELECT
Current_TimeStamp));
ELSE
IF V_Tipo = "D" THEN
DELETE FROM SAPW_Horario_Seminario
WHERE HoSe_Numero = V_HoSe_Numero;
INSERT INTO SAPW_Detalle_Auditoria
VALUES (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
Mantenimiento de Planificación de Certificados se Eliminó el Horario con Código " ||
V_HoSe_Numero::TEXT || " Perteneiente al Seminario " || V_HoSe_Seminario::TEXT
|| " del Certificado " || V_HoSe_Certificado::TEXT, "Se trabajo en la tabla de
Palnificación de Certificados (SAPW_Horario_Seminario)",(SELECT
Current_TimeStamp));
END IF;
END IF;
END IF;
UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion = (SELECT
Current_Date);
RETURN 0;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento Divisiones de la Espol

```

DROP FUNCTION SAPW_Mant_Divisiones(INTEGER,INTEGER, VARCHAR,
VARCHAR, VARCHAR);

```

```
CREATE OR REPLACE FUNCTION SAPW_Mant_Divisiones(INTEGER,INTEGER,
VARCHAR, VARCHAR, VARCHAR) RETURNS INTEGER AS '
```

```
  DECLARE
```

```
    V_Cab_Auditoria      ALIAS FOR $1;
    V_Divi_Codigo        ALIAS FOR $2;
    V_Divi_Nombre        ALIAS FOR $3;
    V_Divi_Descripcion   ALIAS FOR $4;
    V_Tipo               ALIAS FOR $5;
    V_Secuencia          INTEGER;
```

```
  BEGIN
```

```
    IF V_Tipo = "I" THEN
```

```
      SELECT NEXTVAL ("SAPW_S_Division") INTO V_Secuencia;
```

```
      INSERT INTO SAPW_Division VALUES(V_Secuencia,
```

```
V_Divi_Nombre, V_Divi_Descripcion);
```

```
      INSERT INTO SAPW_Detalle_Auditoria VALUES
```

```
(NEXTVAL("SAPW_S_Detalle_Auditoria"), V_Cab_Auditoria, "En el Mantenimiento
de Directivos se Creo el registro con Código " || V_Secuencia::TEXT, "Se trabajo en la
tabla de Divisiones (SAPW_Division)", (SELECT Current_TimeStamp));
```

```
    ELSE
```

```
      V_Secuencia := V_Divi_Codigo;
```

```
      IF V_Tipo = "U" THEN
```

```
        UPDATE SAPW_Division SET Divi_Nombre =
```

```
V_Divi_Nombre, Divi_Descripcion = V_Divi_Descripcion
```

```
WHERE Divi_Codigo = V_Divi_Codigo;
```

```
        INSERT INTO SAPW_Detalle_Auditoria VALUES
```

```
(NEXTVAL("SAPW_S_Detalle_Auditoria"), V_Cab_Auditoria, "En el Mantenimiento
de Directivos se Modificó el registro con Código " || V_Divi_Codigo::TEXT, "Se
trabajo en la tabla de Divisiones (SAPW_Division)", (SELECT Current_TimeStamp));
```

```
      ELSE
```

```
        IF V_Tipo = "D" THEN
```

```
          DELETE FROM SAPW_Division WHERE Divi_Codigo = V_Divi_Codigo;
```

```
          INSERT INTO SAPW_Detalle_Auditoria VALUES
```

```
(NEXTVAL("SAPW_S_Detalle_Auditoria"), V_Cab_Auditoria, "En el
Mantenimiento de Directivos se Eliminó el registro con Código " ||
```

```
V_Divi_Codigo::TEXT, "Se trabajo en la tabla de Divisiones
```

```
(SAPW_Division)", (SELECT Current_TimeStamp));
```

```
        END IF;
```

```
      END IF;
```

```
    END IF;
```

```
    UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
```

```
(SELECT Current_Date);
```

```
    RETURN V_Secuencia;
```

```
  END;
```

```
' LANGUAGE 'plpgsql';
```

```
--Función para el Mantenimiento de Directivos
```

```
DROP FUNCTION SAPW_Mant_Directivos(INTEGER, INTEGER, INTEGER,
INTEGER, VARCHAR, VARCHAR, VARCHAR);
```


CREATE OR REPLACE FUNCTION SAPW_Mant_Directivos(INTEGER,
INTEGER, INTEGER, INTEGER, VARCHAR, VARCHAR, VARCHAR) RETURNS
VOID AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_Dire_Numero	ALIAS FOR \$2;
V_Dire_Division	ALIAS FOR \$3;
V_Dire_Cargo	ALIAS FOR \$4;
V_Dire_Cedula	ALIAS FOR \$5;
V_Dire_Encargado	ALIAS FOR \$6;
V_Tipo	ALIAS FOR \$7;
V_Secuencia	INTEGER;

BEGIN

IF V_Tipo = "I" THEN

SELECT NEXTVAL ("SAPW_S_Directivo") INTO

V_Secuencia;

INSERT INTO SAPW_Directivo VALUES(V_Secuencia,

V_Dire_Division, V_Dire_Cargo, V_Dire_Cedula, V_Dire_Encargado);

INSERT INTO SAPW_Detalle_Auditoria VALUES

(NEXTVAL("SAPW_S_Detalle_Auditoria"), V_Cab_Auditoria, "En el Mantenimiento
de Directivos se Creo el Registro con Código " || V_Secuencia::TEXT || " Perteneciente
a la División " || V_Dire_Division::TEXT,

"Se trabajo en la tabla de Directivos

(SAPW_Directivo)", (SELECT Current_TimeStamp));

ELSE

IF V_Tipo = "U" THEN

UPDATE SAPW_Directivo SET Dire_NombreEncargado

= V_Dire_Encargado, Carg_Codigo = V_Dire_Cargo, Dire_Cedula = V_Dire_Cedula

WHERE Dire_Numero = V_Dire_Numero AND Divi_Codigo = V_Dire_Division;

INSERT INTO SAPW_Detalle_Auditoria VALUES

(NEXTVAL("SAPW_S_Detalle_Auditoria"), V_Cab_Auditoria, "En el Mantenimiento
de Directivos se Modificó el Registro con Código " || V_Dire_Numero::TEXT || "

Perteneciente a la División " || V_Dire_Division::TEXT, "Se trabajo en la tabla de

Directivos (SAPW_Directivo)", (SELECT Current_TimeStamp));

ELSE

IF V_Tipo = "D" THEN

DELETE FROM SAPW_Directivo WHERE

Dire_Numero = V_Dire_Numero AND

Divi_Codigo = V_Dire_Division;

INSERT INTO SAPW_Detalle_Auditoria

VALUES

(NEXTVAL("SAPW_S_Detalle_Auditoria"), V_Ca
b_Auditoria, "En el Mantenimiento de Directivos se
Modificó el Registro con Código " ||

V_Dire_Numero::TEXT || " Perteneciente a la

División " || V_Dire_Division::TEXT, "Se trabajo

en la tabla de Directivos (SAPW_Directivo)",

(SELECT Current_TimeStamp));

END IF;

END IF;




```

        END IF;
        UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
(SELECT Current_Date);
        RETURN 0;
    END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento Documentos Solicitados

```

DROP FUNCTION SAPW_Mant_Documentos(INTEGER,INTEGER, VARCHAR,
VARCHAR);

```

```

CREATE OR REPLACE FUNCTION

```

```

SAPW_Mant_Documentos(INTEGER,INTEGER, VARCHAR, VARCHAR)
RETURNS INTEGER AS '

```

```

    DECLARE

```

```

        V_Cab_Auditoria          ALIAS FOR $1;

```

```

        V_Docu_Codigo            ALIAS FOR $2;

```

```

        V_Docu_Descripcion        ALIAS FOR $3;

```

```

        V_Tipo                    ALIAS FOR $4;

```

```

        V_Secuencia              INTEGER;

```

```

    BEGIN

```

```

        IF V_Tipo = "I" THEN

```

```

            SELECT NEXTVAL ("SAPW_S_Documentos") INTO

```

```

V_Secuencia;

```

```

            INSERT INTO SAPW_Documentos VALUES(V_Secuencia,

```

```

V_Docu_Descripcion);

```

```

            INSERT INTO SAPW_Detalle_Auditoria VALUES

```

```

(NEXTVAL("SAPW_S_Detalle_Auditoria"), V_Cab_Auditoria, "En el Mantenimiento
de Documentos Solicitados para el Diplomado se Creo el registro con Código " ||

```

```

V_Secuencia::TEXT,

```

```

                "Se trabajo en la tabla de Documentos Solicitados
para los Diplomados (SAPW_Documentos)", (SELECT Current_TimeStamp));

```

```

        ELSE

```

```

            V_Secuencia := V_Docu_Codigo;

```

```

            IF V_Tipo = "U" THEN

```

```

                UPDATE SAPW_Documentos SET Docu_Descripcion =

```

```

V_Docu_Descripcion WHERE Docu_Linea = V_Docu_Codigo;

```

```

                INSERT INTO SAPW_Detalle_Auditoria VALUES

```

```

(NEXTVAL("SAPW_S_Detalle_Auditoria"), V_Cab_Auditoria, "En el Mantenimiento
de Documentos Solicitados para los Diplomados se Modificó el registro con Código " ||

```

```

V_Secuencia::TEXT, "Se trabajo en la tabla de Documentos Solicitados para los
Diplomados (SAPW_Documentos)", (SELECT Current_TimeStamp));

```

```

            ELSE

```

```

                IF V_Tipo = "D" THEN

```

```

                    DELETE FROM SAPW_Documentos WHERE Docu_Linea = V_Docu_Codigo;

```

```

                    INSERT INTO SAPW_Detalle_Auditoria VALUES

```

```

(NEXTVAL("SAPW_S_Detalle_Auditoria"), V_Cab_Auditoria, "En el Mantenimiento
de Documentos Solicitados para los Diplomados se Eliminó el registro con Código " ||

```

```

V_Secuencia::TEXT, "Se trabajo en la tabla de Documentos Solicitados para los
Diplomados (SAPW_Documentos)", (SELECT Current_TimeStamp));

```

```

                END IF;

```

```

        END IF;
    END IF;
    UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
(SELECT Current_Date);
    RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Clasificaciones de Foros

DROP FUNCTION

SAPW_Mant_ClasificacionesForos(INTEGER,INTEGER,INTEGER,VARCHAR,VARCHAR,VARCHAR);

CREATE OR REPLACE FUNCTION

SAPW_Mant_ClasificacionesForos(INTEGER,INTEGER,INTEGER,VARCHAR,VARCHAR,VARCHAR) RETURNS VOID AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_Clas_Numero	ALIAS FOR \$2;
V_Clas_Foro	ALIAS FOR \$3;
V_Clas_Nombre	ALIAS FOR \$4;
V_Clas_Descripción	ALIAS FOR \$5;
V_Tipo	ALIAS FOR \$6;
V_Secuencia	INTEGER;

BEGIN

IF V_Tipo = "I" THEN

SELECT NEXTVAL ("SAPW_S_Clasificacion") INTO

V_Secuencia;

INSERT INTO SAPW_Clasificacion VALUES(V_Secuencia,

V_Clas_Foro, V_Clas_Nombre, V_Clas_Descripcion);

INSERT INTO SAPW_Detalle_Auditoria VALUES

(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento de Clasificaciones del Foro se Creo el registro con Código " || V_Secuencia::TEXT || " Perteneciente al Foro con Código " || V_Clas_Foro::TEXT, "Se trabajo en la tabla de Clasificación de Foros de Discusión (SAPW_Clasificacion)",(SELECT Current_TimeStamp));

ELSE

IF V_Tipo = "U" THEN

UPDATE SAPW_Clasificacion SET Clas_Nombre =

V_Clas_Nombre, Clas_Descripcion = V_Clas_Descripcion WHERE Clas_Numero = V_Clas_Numero AND Foru_Numero = V_Clas_Foro;

INSERT INTO SAPW_Detalle_Auditoria VALUES

(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento de Clasificaciones del Foro se Modificó el registro con Código " || Clas_Numero::TEXT || " Perteneciente al Foro con Código " || V_Clas_Foro::TEXT, "Se trabajo en la tabla de Clasificación de Foros de Discusión (SAPW_Clasificacion)",(SELECT Current_TimeStamp));

ELSE

IF V_Tipo = "D" THEN

```

DELETE FROM SAPW_Clasificacion WHERE Clas_Numero = V_Clas_Numero
AND Foru_Numero = V_Clas_Foro;
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Clasificaciones del Foro se Eliminó el registro con Código " ||
V_Clas_Numero::TEXT || " Perteneiente al Foro con Código " || V_Clas_Foro::TEXT,
"Se trabajo en la tabla de Clasificación de Foros de Discusión
(SAPW_Clasificacion)",(SELECT Current_TimeStamp));
END IF;
END IF;
END IF;
RETURN 0;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Preguntas al Foro de Discusión

DROP FUNCTION

SAPW_Mant_PreguntasForo(INTEGER,INTEGER,INTEGER,INTEGER,VARCHAR,
VARCHAR,DATE,VARCHAR,VARCHAR);

CREATE OR REPLACE FUNCTION

SAPW_Mant_PreguntasForo(INTEGER,INTEGER,INTEGER,INTEGER,VARCHAR,
VARCHAR,DATE,VARCHAR,VARCHAR) RETURNS VOID AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_Tema_Numero	ALIAS FOR \$2;
V_Tema_Clasificacion	ALIAS FOR \$3;
V_Tema_Foro	ALIAS FOR \$4;
V_Tema_Pregunta	ALIAS FOR \$5;
V_Tema_Descripción	ALIAS FOR \$6;
V_Tema_Fecha	ALIAS FOR \$7;
V_Tema_Persona	ALIAS FOR \$8;
V_Tipo	ALIAS FOR \$9;
V_Secuencia	INTEGER;

BEGIN

IF V_Tipo = "I" THEN

SELECT NEXTVAL ("SAPW_S_TemaForum") INTO

V_Secuencia;

INSERT INTO SAPW_TemaForum VALUES(V_Secuencia,

V_Tema_Clasificacion, V_Tema_Foro, 1, V_Tema_Pregunta,

V_Tema_Descripcion,

V_Tema_Fecha, 0, V_Tema_Persona);

INSERT INTO SAPW_Detalle_Auditoria VALUES

(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Preguntas al Foro de Discusión se Creo el registro con Código " ||

V_Secuencia::TEXT || " Perteneiente a la Clasificación con código " ||

V_Tema_Clasificacion || " del Foro con Código " || V_Tema_Foro::TEXT,

"Se trabajo en la tabla de Preguntas del Foro

(SAPW_TemaForum)",(SELECT Current_TimeStamp));

ELSE



```

        IF V_Tipo = "U" THEN
UPDATE SAPW_TemaForum SET Tema_Pregunta = V_Tema_Pregunta,
Tema_Descripcion = V_Tema_Descripcion WHERE Tema_Numero =
V_Tema_Numero AND Clas_Numero = V_Tema_Clasificacion AND Foru_Numero =
V_Tema_Foro;
INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Preguntas al Foro de Discusión se Modificó el registro con Código " ||
V_Tema_Numero::TEXT || " Perteneiente a la Clasificación con código " ||
V_Tema_Clasificacion || " del Foro con Código " || V_Tema_Foro::TEXT,
"Se trabajo en la tabla de Preguntas del Foro (SAPW_TemaForum)",(SELECT
Current_TimeStamp));
        ELSE
                IF V_Tipo = "D" THEN
                        DELETE FROM SAPW_TemaForum WHERE
Tema_Numero = V_Tema_Numero AND Clas_Numero = V_Tema_Clasificacion AND
Foru_Numero = V_Tema_Foro;
                        INSERT INTO SAPW_Detalle_Auditoria
VALUES (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
Mantenimiento de Preguntas al Foro de Discusión se Eliminó el registro con Código " ||
V_Tema_Numero::TEXT || " Perteneiente a la Clasificación con código " ||
V_Tema_Clasificacion || " del Foro con Código " || V_Tema_Foro::TEXT,
"Se trabajo en la tabla de Preguntas del Foro (SAPW_TemaForum)",(SELECT
Current_TimeStamp));
                END IF;
        END IF;
        RETURN 0;
END;
' LANGUAGE 'plpgsql';

```

--Función para el Mantenimiento de Respuestas al Foro de Discusión

DROP FUNCTION

SAPW_Mant_RespuestasForo(INTEGER,INTEGER,INTEGER,INTEGER,INTEGER,
VARCHAR,VARCHAR,DATE,VARCHAR);

CREATE OR REPLACE FUNCTION

SAPW_Mant_RespuestasForo(INTEGER,INTEGER,INTEGER,INTEGER,INTEGER,
VARCHAR,VARCHAR,DATE,VARCHAR) RETURNS VOID AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_Resp_Numero	ALIAS FOR \$2;
V_Resp_Tema	ALIAS FOR \$3;
V_Resp_Clasificacion	ALIAS FOR \$4;
V_Resp_Foro	ALIAS FOR \$5;
V_Resp_Persona	ALIAS FOR \$6;
V_Resp_Respuesta	ALIAS FOR \$7;
V_Tema_Fecha	ALIAS FOR \$8;
V_Tipo	ALIAS FOR \$9;
V_Secuencia	INTEGER;

```

BEGIN
    IF V_Tipo = "I" THEN
        SELECT NEXTVAL ("SAPW_S_Respuesta") INTO
V_Secuencia;
        INSERT INTO SAPW_Respuesta VALUES(V_Secuencia,
V_Resp_Tema, V_Resp_Clasificacion, V_Resp_Foro, V_Resp_Persona, 1,
V_Resp_Respuesta, V_Tema_Fecha);
        INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Respuestas al Foro de Discusión se Creo el registro con Código " ||
V_Secuencia::TEXT || " Perteneiente a la Pregunta" || V_Resp_Tema || " de la
Clasificación con código " || V_Resp_Clasificacion || " del Foro con Código " ||
V_Resp_Foro::TEXT,
"Se trabajo en la tabla de Respuestas del Foro
(SAPW_Respuestas)",(SELECT Current_TimeStamp));
    ELSE
        IF V_Tipo = "U" THEN
            UPDATE SAPW_Respuesta SET Resp_Respuesta =
V_Resp_Respuesta
            WHERE Resp_Numero =
V_Resp_Numero AND Tema_Numero = V_Resp_Tema AND Clas_Numero =
V_Resp_Clasificacion AND Foru_Numero = V_Resp_Foro;
            INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Respuestas al Foro de Discusión se Modificó el registro con Código " ||
V_Resp_Numero::TEXT || " Perteneiente a la Pregunta" || V_Resp_Tema || " de la
Clasificación con código " || V_Resp_Clasificacion || " del Foro con Código " ||
V_Resp_Foro::TEXT,
"Se trabajo en la tabla de Respuestas del Foro
(SAPW_Respuestas)",(SELECT Current_TimeStamp));
        ELSE
            IF V_Tipo = "D" THEN
                DELETE FROM SAPW_Respuesta WHERE
Resp_Numero = V_Resp_Numero AND Tema_Numero = V_Resp_Tema AND
Clas_Numero = V_Resp_Clasificacion AND Foru_Numero = V_Resp_Foro;
                INSERT INTO SAPW_Detalle_Auditoria
VALUES (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
Mantenimiento de Respuestas al Foro de Discusión se Eliminó el registro con Código " ||
V_Resp_Numero::TEXT || " Perteneiente a la Pregunta" || V_Resp_Tema || " de la
Clasificación con código " || V_Resp_Clasificacion || " del Foro con Código " ||
V_Resp_Foro::TEXT,
"Se trabajo en la tabla de Respuestas
del Foro (SAPW_Respuestas)",(SELECT Current_TimeStamp));
            END IF;
        END IF;
    END IF;
    RETURN 0;
END;
' LANGUAGE 'plpgsql';

```


--Función para Verificar si es que esta Relacionado el Laboratorio con la Tabla Horarios por Paralelo

```
DROP FUNCTION SAPW_ExisteLaboratorio(INTEGER);
CREATE OR REPLACE FUNCTION SAPW_ExisteLaboratorio(INTEGER)
RETURNS BOOLEAN AS '
    DECLARE
        V_Labo_Codigo ALIAS FOR $1;
        V_Registro RECORD;
    BEGIN
        SELECT INTO V_Registro * FROM SAPW_Horario_Paralelo WHERE
Tipo_Codigo = 27 AND HoPa_Aula = V_Labo_Codigo;
        IF NOT FOUND THEN
            RETURN FALSE;
        ELSE
            RETURN TRUE;
        END IF;
    END;
' LANGUAGE 'plpgsql';
```

--Función para Verificar si es que esta Relacionado el Aula de Clase con la Tabla Horarios por Paralelo

```
DROP FUNCTION SAPW_ExisteAula(INTEGER);
CREATE OR REPLACE FUNCTION SAPW_ExisteAula(INTEGER) RETURNS
BOOLEAN AS '
    DECLARE
        V_SaCl_Codigo ALIAS FOR $1;
        V_Registro RECORD;
    BEGIN
        SELECT INTO V_Registro * FROM SAPW_Horario_Paralelo WHERE
Tipo_Codigo = 26 AND HoPa_Aula = V_SaCl_Codigo;
        IF NOT FOUND THEN
            RETURN FALSE;
        ELSE
            RETURN TRUE;
        END IF;
    END;
' LANGUAGE 'plpgsql';
```

--Función para Verificar la cantidad de archivos a eliminar del servidor

```
DROP FUNCTION SAPW_CantidadArchivos();
CREATE OR REPLACE FUNCTION SAPW_CantidadArchivos() RETURNS
INTEGER AS '
    DECLARE
        V_Cantidad INTEGER;
    BEGIN
        SELECT COUNT(*) INTO V_Cantidad FROM SAPW_BackupOnLine
WHERE ((SELECT Current_TimeStamp) - Baon_Fecha) >= 2;
        RETURN V_Cantidad;
```




```

END;
' LANGUAGE 'plpgsql';

--Función para Mantenimiento de Becas
DROP FUNCTION SAPW_mant_beca(int4, int4, varchar, varchar, varchar, varchar)
CREATE FUNCTION public.sapw_mant_beca(int4, int4, varchar, varchar, varchar,
varchar) RETURNS int4 AS '
    DECLARE
        V_Cab_Auditoria          ALIAS FOR $1;
        V_Codigo                 ALIAS FOR $2;
        V_titulo                 ALIAS FOR $3;
        V_descripcion            ALIAS FOR $4;
        V_objetivo               ALIAS FOR $5;
        V_Tipo                   ALIAS FOR $6;
        V_Secuencia              INTEGER;
    BEGIN
        IF V_Tipo = "I" THEN
            SELECT NEXTVAL ("SAPW_S_Beca") INTO V_Secuencia;
            INSERT INTO SAPW_Fondo_Beca VALUES(V_Secuencia,
V_titulo, V_descripcion, V_objetivo);
            INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Becas se Creo el registro con Código " || V_Secuencia::TEXT, "Se trabajo en la tabla
de Becas (SAPW_Fondo_beca)",(SELECT Current_TimeStamp));
        ELSE
            V_Secuencia := V_Codigo;
            IF V_Tipo = "U" THEN
                UPDATE SAPW_Fondo_beca SET fobe_titulo =
V_titulo, fobe_descripcion = V_descripcion, fobe_objetivo = V_objetivo
                WHERE fobe_Codigo = V_Codigo;
                INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Becas se Modificó el registro con Código " || V_Codigo::TEXT, "Se trabajo en la
tabla de Becas (SAPW_Fondo_Beca)",(SELECT Current_TimeStamp));
            ELSE
                IF V_Tipo = "D" THEN
                    DELETE FROM SAPW_Fondo_Beca WHERE
fobe_Codigo = V_Codigo;
                    INSERT INTO SAPW_Detalle_Auditoria
VALUES (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
Mantenimiento de Becas se Eliminó el registro con Código " || V_Codigo::TEXT, "Se
trabajo en la tabla de Becas (SAPW_Fondo_Beca)",(SELECT Current_TimeStamp));
                END IF;
            END IF;
        END IF;
        RETURN V_Secuencia;
    END;
' LANGUAGE 'plpgsql';

```

--Función para Mantenimiento de Cabecera de Testimonios

DROP FUNCTION SAPW_mant_cabeceratestimonios(int4, int4, varchar, varchar, varchar, varchar, varchar, varchar, int4, varchar)

CREATE FUNCTION public.sapw_mant_cabeceratestimonios(int4, int4, varchar, varchar, varchar, varchar, varchar, varchar, int4, varchar) RETURNS int4 AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_Test_Codigo	ALIAS FOR \$2;
V_Test_Nombre	ALIAS FOR \$3;
V_Test_Titulo	ALIAS FOR \$4;
V_Test_Empresa	ALIAS FOR \$5;
V_Test_Cargo	ALIAS FOR \$6;
V_Test_Comentario	ALIAS FOR \$7;
V_Test_Tiempo	ALIAS FOR \$8;
V_Test_Estado	ALIAS FOR \$9;
V_Tipo	ALIAS FOR \$10;
V_Secuencia	INTEGER;

BEGIN

IF V_Tipo = "I" THEN

SELECT NEXTVAL ("SAPW_S_Testimonios") INTO

V_Secuencia;

INSERT INTO SAPW_Testimonios VALUES(V_Secuencia,
V_Test_Nombre, V_Test_Titulo, V_Test_Cargo, V_Test_Comentario, NEXTVAL
("SAPW_S_Foto_Cabecera"), V_Test_Empresa, NEXTVAL
("SAPW_S_Foto_Cabecera"), V_Test_Tiempo, V_Test_Estado, NEXTVAL
("SAPW_S_Foto_Cabecera"), NEXTVAL ("SAPW_S_Foto_Cabecera"), NEXTVAL
("SAPW_S_Foto_Cabecera"));

INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"), V_Cab_Auditoria, "En el Mantenimiento
de Testimonios se Creo el registro con Código " || V_Secuencia::TEXT, "Se trabajo en
la tabla de Testimonios (SAPW_Testimonios)", (SELECT Current_TimeStamp));

ELSE

V_Secuencia := V_Test_Codigo;

IF V_Tipo = "U" THEN

UPDATE SAPW_Testimonios SET Test_Nombre =
V_Test_Nombre, Test_Titulo = V_Test_Titulo, Test_Cargo=V_Test_Cargo,
Test_Comentario= V_Test_Comentario, Test_Empresa=V_Test_Empresa,
Test_Tiempo=V_Test_Tiempo, Esta_Codigo=V_Test_Estado WHERE Test_Numero =
V_Test_Codigo;

INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"), V_Cab_Auditoria, "En el Mantenimiento
de Testimonios se Modificó el registro con Código " || V_Test_Codigo::TEXT, "Se
trabajo en la tabla de Testimonios (SAPW_Testimonios)", (SELECT
Current_TimeStamp));

ELSE

IF V_Tipo = "D" THEN

DELETE FROM SAPW_Testimonios WHERE
Test_Numero = V_Test_Codigo;



```

        INSERT INTO SAPW_Detalle_Auditoria
VALUES (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
Mantenimiento de Testimonios se Eliminó el registro con Código " ||
V_Test_Codigo::TEXT, "Se trabajo en la tabla de Testimonios (SAPW_Testimonios)",
(SELECT Current_TimeStamp));
        END IF;
    END IF;
END IF;
UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
(SELECT Current_Date);
RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

```

--Función para Mantenimiento de Detalle de Beca

DROP FUNCTION SAPW_mant_detalle_beca(int4, int4, int4, varchar, varchar,
varchar, int4, varchar)

CREATE FUNCTION public.sapw_mant_detalle_beca(int4, int4, int4, varchar,
varchar, varchar, int4, varchar) RETURNS int4 AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_det_Codigo	ALIAS FOR \$2;
V_det_codigo1	ALIAS FOR \$3;
V_det_cobertura	ALIAS FOR \$4;
V_det_beneficiario	ALIAS FOR \$5;
V_det_beca	ALIAS FOR \$6;
V_esta_codigo	ALIAS FOR \$7;
V_Tipo	ALIAS FOR \$8;
V_secuencia	INTEGER;

BEGIN

IF V_Tipo = "I" THEN

SELECT NEXTVAL ("SAPW_S_detalle_beca") INTO

V_Secuencia;

INSERT INTO SAPW_Detalle_fondo_beca

VALUES(V_secuencia, V_det_codigo1, V_det_cobertura, V_det_beneficiario,
V_det_beca, V_esta_codigo);

ELSE

V_Secuencia := V_det_Codigo;

IF V_Tipo = "U" THEN

UPDATE SAPW_detalle_fondo_beca SET

debe_cobertura = V_det_cobertura, debe_beneficiario = V_det_beneficiario, debe_beca
= V_det_beca, esta_codigo = V_esta_codigo WHERE debe_Codigo = V_det_Codigo
AND fobe_codigo = V_det_codigo1;

ELSE

IF V_Tipo = "D" THEN

DELETE FROM SAPW_detalle_fondo_beca

WHERE debe_Codigo = V_det_Codigo AND fobe_codigo = V_det_codigo1;

END IF;

END IF;

```

        END IF;
        UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
(SELECT Current_Date);
        RETURN V_Secuencia;
    END;
' LANGUAGE 'plpgsql';

```

--Función para Mantenimiento de Detalle de Testimonios

```

DROP FUNCTION SAPW_mant_detalletestimonios(int4, int4, int4, int4, varchar, int4,
varchar)

```

```

CREATE FUNCTION public.sapw_mant_detalletestimonios(int4, int4, int4, int4,
varchar, int4, varchar) RETURNS void AS '

```

```

    DECLARE

```

```

        V_Cab_Auditoria          ALIAS FOR $1;
        V_Pregunta               ALIAS FOR $2;
        V_Test_Numero            ALIAS FOR $3;
        V_Preg_Numero            ALIAS FOR $4;
        V_Det_Respuesta          ALIAS FOR $5;
        V_Det_Estado             ALIAS FOR $6;
        V_Tipo                   ALIAS FOR $7;

```

```

    BEGIN

```

```

        IF V_Tipo = "I" THEN

```

```

            INSERT INTO SAPW_Detalle_Testimonios VALUES
(V_Test_Numero, V_Preg_Numero, V_Det_Respuesta, V_Det_Estado);
            INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"), V_Cab_Auditoria,"En el Mantenimiento
de Detalle de Testimonios se agregó el Registro que contiene la Pregunta con Código "
|| V_Preg_Numero || " Pertenece al Testimonio con Código " || V_Test_Numero,
"Se trabajo en la tabla de Detalle de Testimonios
(SAPW_Detalle_Testimonios)",(SELECT Current_TimeStamp));

```

```

        ELSE

```

```

            IF V_Tipo = "U" THEN

```

```

                UPDATE SAPW_Detalle_Testimonios SET
Preg_Numero = V_Preg_Numero, Det_Respuesta=V_Det_Respuesta,
Esta_Codigo=V_Det_Estado WHERE Test_Numero = V_Test_Numero and
Preg_Numero = V_Pregunta;

```

```

                INSERT INTO SAPW_Detalle_Auditoria VALUES
(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Detalle de Testimonios se Modificó el Registro que contiene la Pregunta con Código
" || V_Pregunta || " Pertenece al Testimonio con Código " || V_Test_Numero,
"Se trabajo en la tabla de Detalle de Testimonios (SAPW_Detalle_Testimonios)",
(SELECT Current_TimeStamp));

```

```

            ELSE

```

```

                IF V_Tipo = "D" THEN

```

```

                    DELETE FROM SAPW_Detalle_Testimonios
WHERE Test_Numero = V_Test_Numero AND Preg_Numero = V_Pregunta;
                    INSERT INTO SAPW_Detalle_Auditoria
VALUES (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
Mantenimiento de Detalle de Testimonios se Eliminó el Resgistro que contiene la

```

```

Pregunta con Código " || V_Pregunta || " Pertenece al Testimonio con Código " ||
V_Test_Numero, "Se trabajo en la tabla de Detalle de Testimonios
(SAPW_Detalle_Testimonios)", (SELECT Current_TimeStamp));
        END IF;
    END IF;
END IF;
UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
(SELECT Current_Date);
RETURN 0;
END;
' LANGUAGE 'plpgsql';

```

--Función para Mantenimiento de Preguntas de Testimonios

DROP FUNCTION SAPW_mant_preguntastestimonios(int4, int4, varchar, int4, varchar)

CREATE FUNCTION public.sapw_mant_preguntastestimonios(int4, int4, varchar, int4, varchar) RETURNS int4 AS '

DECLARE

V_Cab_Auditoria	ALIAS FOR \$1;
V_Preg_Codigo	ALIAS FOR \$2;
V_Preg_Pregunta	ALIAS FOR \$3;
V_Estado	ALIAS FOR \$4;
V_Tipo	ALIAS FOR \$5;
V_Secuencia	INTEGER;

BEGIN

IF V_Tipo = "I" THEN

SELECT NEXTVAL ("sapw_s_preguntas_testimonios ") INTO

V_Secuencia;

INSERT INTO SAPW_Preguntas_Testimonios

VALUES(V_Secuencia, V_Preg_Pregunta, V_Estado);

INSERT INTO SAPW_Detalle_Auditoria VALUES

(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento de Preguntas de Testimonios se Creo el registro con Código " || V_Secuencia::TEXT, "Se trabajo en la tabla de Preguntas de Testimonios

(SAPW_Preguntas_Testimonios)",(SELECT Current_TimeStamp));

ELSE

V_Secuencia := V_Preg_Codigo ;

IF V_Tipo = "U" THEN

UPDATE SAPW_Preguntas_Testimonios SET

Preg_Pregunta = V_Preg_Pregunta, Esta_Codigo= V_Estado WHERE Preg_Numero = V_Preg_Codigo;

INSERT INTO SAPW_Detalle_Auditoria VALUES

(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento de Preguntas de Testimonios se Modificó el registro con Código " ||

V_Preg_Codigo::TEXT, "Se trabajo en la tabla de Preguntas de Testimonios

(SAPW_Preguntas_Testimonios)",(SELECT Current_TimeStamp));

ELSE

IF V_Tipo = "D" THEN




```

DELETE FROM SAPW_Preguntas_Testimonios
WHERE Preg_numero = V_Preg_Codigo;
INSERT INTO SAPW_Detalle_Auditoria
VALUES (NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el
Mantenimiento de Preguntas de Testimonios se Eliminó el registro con Código " ||
V_Preg_Codigo::TEXT, "Se trabajo en la tabla de Preguntas de Testimonios
(SAPW_Preguntas_Testimonios)",(SELECT Current_TimeStamp));
END IF;
END IF;
END IF;
UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
(SELECT Current_Date);
RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

```

--Función para Mantenimiento de Requisito Beca

```

DROP FUNCTION SAPW_mant_requisitobeca(int4, int4, int4, varchar, varchar)
CREATE FUNCTION public.sapw_mant_requisitobeca(int4, int4, int4, varchar,
varchar) RETURNS int4 AS '

```

```

DECLARE

```

```

    V_Cab_Auditoria      ALIAS FOR $1;
    V_Rebe_Codigo        ALIAS FOR $2;
    V_fobe_codigo        ALIAS FOR $3;
    V_rebe_Descripcion    ALIAS FOR $4;
    V_Tipo                ALIAS FOR $5;
    V_Secuencia           INTEGER;

```

```

BEGIN

```

```

    IF V_Tipo = "I" THEN

```

```

        SELECT NEXTVAL ("SAPW_S_Requisito_Beca") INTO

```

```

V_Secuencia;

```

```

        INSERT INTO SAPW_Requisito_Beca VALUES(V_Secuencia,

```

```

V_fobe_Codigo, V_rebe_Descripcion);

```

```

        INSERT INTO SAPW_Detalle_Auditoria VALUES

```

```

(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Fondo Beca se Creo el registro con Código " || V_Secuencia::TEXT || " Perteneiente
al Fondo Beca con Código " || V_fobe_Codigo::TEXT, "Se trabajo en la tabla de
Requisito Beca (SAPW_Requisito_Beca)",(SELECT Current_TimeStamp));

```

```

    ELSE

```

```

        V_Secuencia := V_rebe_Codigo;

```

```

        IF V_Tipo = "U" THEN

```

```

            UPDATE SAPW_Requisito_Beca SET rebe_Descripcion
= V_rebe_Descripcion WHERE fobe_Codigo = V_fobe_Codigo AND rebe_Codigo =
V_rebe_Codigo;

```

```

            INSERT INTO SAPW_Detalle_Auditoria VALUES

```

```

(NEXTVAL("SAPW_S_Detalle_Auditoria"),V_Cab_Auditoria,"En el Mantenimiento
de Fondo Beca se Modificó el registro con Código " || V_rebe_Codigo::TEXT || "
Perteneiente al Fondo Beca con Código " || V_fobe_Codigo::TEXT, "Se trabajo en la
tabla de Requisito Beca (SAPW_Requisito_Beca)",(SELECT Current_TimeStamp));

```



```

ELSE
    IF V_Tipo = "D" THEN
        DELETE FROM SAPW_Requisito_Beca WHERE
fobe_Codigo = V_fobe_Codigo AND rebe_Codigo = V_rebe_Codigo;
        INSERT INTO SAPW_Detalle_Auditoria
VALUES (NEXTVAL("SAPW_S_Detalle_Auditoria"), V_Cab_Auditoria, "En el
Mantenimiento de Fondo Beca se Eliminó el registro con Código " ||
V_rebe_Codigo::TEXT || " Perteneiente al Fondo Beca con Código " ||
V_fobe_codigo::TEXT, "Se trabajo en la tabla de Requisito Beca
(SAPW_Requisito_Beca)", (SELECT Current_TimeStamp));
        END IF;
    END IF;
    UPDATE SAPW_Parametro SET Para_FechaUltimaActualizacion =
(SELECT Current_Date);
    RETURN V_Secuencia;
END;
' LANGUAGE 'plpgsql';

```

4.1.7 Creación de Desencadenadores.

```

-----
CREATE OR REPLACE FUNCTION On_Update_RolWeb() RETURNS TRIGGER
AS '
    BEGIN
        IF NEW.Esta_Codigo <> OLD.Esta_Codigo THEN
            UPDATE SAPW_Perfil_Web SET Esta_Codigo =
NEW.Esta_Codigo
            WHERE OpWe_Codigo = NEW.OpWe_Codigo AND
Usua_Login IN (SELECT Usua_Login FROM SAPW_Usuario WHERE GrUs_Codigo
= NEW.GrUs_Codigo);
            END IF;
            RETURN NEW;
        END;'
LANGUAGE 'plpgsql';

--DROP TRIGGER SAPW_TU_Rol_Web ON SAPW_Rol_Web;
CREATE TRIGGER SAPW_TU_Rol_Web BEFORE UPDATE
ON SAPW_Rol_Web FOR EACH ROW
EXECUTE PROCEDURE On_Update_RolWeb();

```

```

-----
CREATE OR REPLACE FUNCTION On_Update_RolSistema() RETURNS
TRIGGER AS '
    BEGIN
        IF NEW.Esta_Codigo <> OLD.Esta_Codigo THEN

```

```

        UPDATE SAPW_Perfil_Usuario SET Esta_Codigo =
NEW.Esta_Codigo
        WHERE OpSi_Codigo = NEW.OpSi_Codigo AND Usua_Login
IN (SELECT Usua_Login FROM SAPW_Usuario WHERE GrUs_Codigo =
NEW.GrUs_Codigo);
        END IF;
        RETURN NEW;
    END;'
LANGUAGE 'plpgsql';

--DROP TRIGGER SAPW_TU_Rol_Sistema ON SAPW_Rol_Sistema;
CREATE TRIGGER SAPW_TU_Rol_Sistema BEFORE UPDATE
    ON SAPW_Rol_Sistema FOR EACH ROW
EXECUTE PROCEDURE On_Update_RolSistema();

```

```

CREATE OR REPLACE FUNCTION On_Insert_GrupoUsuario() RETURNS
TRIGGER AS '
    DECLARE
        V_Opcion RECORD;
        V_Valor      INTEGER;
    BEGIN
        IF ((NEW.GrUs_Codigo = 0) OR (NEW.GrUs_Codigo = 5)) THEN
            V_Valor := 1;
        ELSE
            V_Valor := 2;
        END IF;
        IF NEW.GrUs_Pertenece = 1 THEN
            FOR V_Opcion IN SELECT * FROM SAPW_Opcion_Sistema
LOOP
                INSERT INTO SAPW_Rol_Sistema VALUES
(NEW.GrUs_Codigo, V_Opcion.OpSi_Codigo, V_Opcion.OpSi_OpcionCodigo,
V_Valor);
            END LOOP;
        ELSE
            FOR V_Opcion IN SELECT * FROM SAPW_Opcion_Web
LOOP
                INSERT INTO SAPW_Rol_Web VALUES
(NEW.GrUs_Codigo, V_Opcion.OpWe_Codigo, V_Opcion.OpWe_OpcionCodigo,
V_Valor);
            END LOOP;
        END IF;
        RETURN NEW;
    END;'
LANGUAGE 'plpgsql';

--DROP TRIGGER SAPW_TI_Grupos ON SAPW_Grupo_Usuario;
CREATE TRIGGER SAPW_TI_Grupos BEFORE INSERT
    ON SAPW_Grupo_Usuario FOR EACH ROW

```

```
EXECUTE PROCEDURE On_Insert_GrupoUsuario();
```

```
-----  
CREATE OR REPLACE FUNCTION On_Insert_Producto() RETURNS TRIGGER  
AS '  
    BEGIN  
        INSERT INTO SAPW_FOTO VALUES (NEW.Foto_Codigo, 0, NULL,  
NULL, NULL, NULL, 2);  
        RETURN NEW;  
    END;'  
LANGUAGE 'plpgsql';
```

```
--DROP TRIGGER SAPW_TI_Productos ON SAPW_Producto;  
CREATE TRIGGER SAPW_TI_Productos BEFORE INSERT  
    ON SAPW_Producto FOR EACH ROW  
    EXECUTE PROCEDURE On_Insert_Producto();
```

```
-----  
CREATE OR REPLACE FUNCTION On_Delete_Producto() RETURNS TRIGGER  
AS '  
    BEGIN  
        DELETE FROM SAPW_FOTO WHERE Foto_Codigo =  
OLD.Foto_Codigo AND Foto_Numero = 0;  
        RETURN OLD;  
    END;'  
LANGUAGE 'plpgsql';
```

```
--DROP TRIGGER SAPW_TD_Productos ON SAPW_Producto;  
CREATE TRIGGER SAPW_TD_Productos BEFORE DELETE  
    ON SAPW_Producto FOR EACH ROW  
    EXECUTE PROCEDURE On_Delete_Producto();
```

```
-----  
CREATE OR REPLACE FUNCTION On_Insert_Estudiante() RETURNS TRIGGER  
AS '  
    DECLARE  
        V_Persona RECORD;  
    BEGIN  
        SELECT INTO V_Persona * FROM SAPW_Persona WHERE  
Pers_Cedula = NEW.Estu_Cedula;  
        IF NOT FOUND THEN  
            INSERT INTO SAPW_Persona VALUES (NEW.Estu_Cedula, 1,  
11, NEW.Estu_Apellidos, NEW.Estu_Nombres, NEW.Estu_FechaNacimiento,  
NEW.Estu_Direccion,  
NEW.Estu_Telefono, NEW.Estu_Email, NEW.Estu_EstadoCivil, NEW.Estu_Sexo,  
NEXTVAL("SAPW_S_Foto_Cabecera"), 3,  
NEW.Tipo_Identificacion, NEW.Esta_Codigo, NULL);  
        ELSE
```

```
UPDATE SAPW_Persona SET Tipo_Codigo = 11,
Pers_Apellidos = NEW.Estu_Apellidos, Pers_Nombres = NEW.Estu_Nombres,
                                Pers_FechaNacimiento =
NEW.Estu_FechaNacimiento, Pers_Direccion = NEW.Estu_Direccion,
                                Pers_Telefono = NEW.Estu_Telefono,
Pers_Email = NEW.Estu_Email,
                                Pers_EstadoCivil =
NEW.Estu_EstadoCivil, Pers_Sexo = NEW.Estu_Sexo,
                                Tipo_Identificacion =
NEW.Tipo_Identificacion, Esta_Codigo = NEW.Esta_Codigo
                                WHERE Pers_Cedula = NEW.Estu_Cedula;
END IF;
RETURN NEW;
END;'
LANGUAGE 'plpgsql';
```

```
--DROP TRIGGER SAPW_TI_Estudiantes ON SAPW_Estudiante;
CREATE TRIGGER SAPW_TI_Estudiantes BEFORE INSERT
ON SAPW_Estudiante FOR EACH ROW
EXECUTE PROCEDURE On_Insert_Estudiante();
```

CREATE OR REPLACE FUNCTION On_Update_Estudiante() RETURNS TRIGGER
AS '

```
BEGIN
UPDATE SAPW_Persona SET Pers_Apellidos = NEW.Estu_Apellidos,
Pers_Nombres = NEW.Estu_Nombres,
                                Pers_FechaNacimiento =
NEW.Estu_FechaNacimiento, Pers_Direccion = NEW.Estu_Direccion,
                                Pers_Telefono = NEW.Estu_Telefono, Pers_Email
= NEW.Estu_Email,
                                Pers_EstadoCivil = NEW.Estu_EstadoCivil,
Pers_Sexo = NEW.Estu_Sexo,
                                Tipo_Identificacion = NEW.Tipo_Identificacion,
Esta_Codigo = NEW.Esta_Codigo
                                WHERE Pers_Cedula = NEW.Estu_Cedula;
RETURN NEW;
END;'
LANGUAGE 'plpgsql';
```

```
--DROP TRIGGER SAPW_TU_Estudiantes ON SAPW_Estudiante;
CREATE TRIGGER SAPW_TU_Estudiantes BEFORE UPDATE
ON SAPW_Estudiante FOR EACH ROW
EXECUTE PROCEDURE On_Update_Estudiante();
```

```
-----
CREATE OR REPLACE FUNCTION On_Delete_Estudiante() RETURNS TRIGGER
AS '
    BEGIN
        DELETE FROM SAPW_Persona WHERE Pers_Cedula =
OLD.Estu_Cedula;
        RETURN OLD;
    END;'
LANGUAGE 'plpgsql';

--DROP TRIGGER SAPW_TD_Estudiantes ON SAPW_Estudiante;
CREATE TRIGGER SAPW_TD_Estudiantes BEFORE DELETE
    ON SAPW_Estudiante FOR EACH ROW
    EXECUTE PROCEDURE On_Delete_Estudiante();
```

4.1.8 Generación de Datos Constantes.

--Estados de Animo

```
INSERT INTO SAPW_Estado_Animo VALUES (1, 'Defalut', 0);
INSERT INTO SAPW_Estado_Animo VALUES (2, 'Tenso', 0);
INSERT INTO SAPW_Estado_Animo VALUES (3, 'Emprendedor', 0);
INSERT INTO SAPW_Estado_Animo VALUES (4, 'Eufórico', 0);
INSERT INTO SAPW_Estado_Animo VALUES (5, 'Sensible', 0);
INSERT INTO SAPW_Estado_Animo VALUES (6, 'Relajado', 0);
```

--Colores de los Estados de Animo

```
INSERT INTO SAPW_Colores_Animo VALUES ( 1, 1, '#EDF5FA', 'Celeste Claro');
INSERT INTO SAPW_Colores_Animo VALUES ( 2, 1, '#628F9F', 'Nuevo Usuario');
INSERT INTO SAPW_Colores_Animo VALUES ( 3, 1, '#8F9B45', 'Olvido su
Contraseña');
INSERT INTO SAPW_Colores_Animo VALUES ( 4, 1, '#4D4D4D', 'Franja');
INSERT INTO SAPW_Colores_Animo VALUES ( 5, 1, '#ABAC8D', 'Noticias TI');
INSERT INTO SAPW_Colores_Animo VALUES ( 6, 1, '#45BA9F', 'Letras de
Servicios y Productos');
INSERT INTO SAPW_Colores_Animo VALUES ( 7, 1, '#4E555D', 'Franja de
Servicios y Productos');
INSERT INTO SAPW_Colores_Animo VALUES ( 8, 1, '#209CEF', 'Celeste');
INSERT INTO SAPW_Colores_Animo VALUES ( 9, 1, '#FF6300', 'Anaranjado');
INSERT INTO SAPW_Colores_Animo VALUES (10, 1, '#79C517', 'Verde');
INSERT INTO SAPW_Colores_Animo VALUES (11, 1, '#CB3398', 'Lila');
INSERT INTO SAPW_Colores_Animo VALUES (12, 1, '#989A62', 'Cafe');
```

--Estados

```
INSERT INTO SAPW_Estado VALUES( 1, 'Activo', 1);
INSERT INTO SAPW_Estado VALUES( 2, 'Inactivo', 1);
INSERT INTO SAPW_Estado VALUES( 3, 'Leido', 2);
INSERT INTO SAPW_Estado VALUES( 4, 'No leido', 2);
INSERT INTO SAPW_Estado VALUES( 5, 'Aprobado', 3);
INSERT INTO SAPW_Estado VALUES( 6, 'No Aprobado', 3);
```



```
INSERT INTO SAPW_Estado VALUES( 7, 'Abierto', 4);
INSERT INTO SAPW_Estado VALUES( 8, 'Cerrado', 4);
INSERT INTO SAPW_Estado VALUES( 9, 'Disponible', 5);
INSERT INTO SAPW_Estado VALUES(10, 'No Disponible', 5);
INSERT INTO SAPW_Estado VALUES(11, 'Publicado', 6);
INSERT INTO SAPW_Estado VALUES(12, 'No Publicado', 6);
INSERT INTO SAPW_Estado VALUES(13, 'En Espera', 3);
```

--Tipos

```
INSERT INTO SAPW_Tipo VALUES( 1, 'Carrera Regular', 1);
INSERT INTO SAPW_Tipo VALUES( 2, 'Carrera Técnica', 1);
INSERT INTO SAPW_Tipo VALUES( 3, 'Interno', 2);
INSERT INTO SAPW_Tipo VALUES( 4, 'Externo', 2);
INSERT INTO SAPW_Tipo VALUES( 5, 'Teórica', 3);
INSERT INTO SAPW_Tipo VALUES( 6, 'Teórica/Práctica', 3);
INSERT INTO SAPW_Tipo VALUES( 7, 'Práctica', 3);
INSERT INTO SAPW_Tipo VALUES( 8, 'Noticias', 4);
INSERT INTO SAPW_Tipo VALUES( 9, 'Sitios', 4);
INSERT INTO SAPW_Tipo VALUES(10, 'Profesor', 5);
INSERT INTO SAPW_Tipo VALUES(11, 'Estudiante', 5);
INSERT INTO SAPW_Tipo VALUES(12, 'Empresario', 5);
INSERT INTO SAPW_Tipo VALUES(13, 'Ex-Alumno', 5);
INSERT INTO SAPW_Tipo VALUES(14, 'Temporal', 6);
INSERT INTO SAPW_Tipo VALUES(15, 'Permanente', 6);
INSERT INTO SAPW_Tipo VALUES(16, 'Perfil Ocupacional', 7);
INSERT INTO SAPW_Tipo VALUES(17, 'Perfil Profesional', 7);
INSERT INTO SAPW_Tipo VALUES(18, 'Parcial', 8);
INSERT INTO SAPW_Tipo VALUES(19, 'Final', 8);
INSERT INTO SAPW_Tipo VALUES(20, 'Mejoramiento', 8);
INSERT INTO SAPW_Tipo VALUES(21, 'Deberes', 9);
INSERT INTO SAPW_Tipo VALUES(22, 'Lecciones', 9);
INSERT INTO SAPW_Tipo VALUES(23, 'Proyecto', 9);
INSERT INTO SAPW_Tipo VALUES(24, 'Examen', 9);
INSERT INTO SAPW_Tipo VALUES(25, 'Otros', 9);
INSERT INTO SAPW_Tipo VALUES(26, 'Aulas', 10);
INSERT INTO SAPW_Tipo VALUES(27, 'Laboratorios', 10);
INSERT INTO SAPW_Tipo VALUES(28, 'Individual', 11);
INSERT INTO SAPW_Tipo VALUES(29, 'Completo', 11);
INSERT INTO SAPW_Tipo VALUES(30, 'Sistema', 12);
INSERT INTO SAPW_Tipo VALUES(31, 'CD-ROM', 12);
INSERT INTO SAPW_Tipo VALUES(32, 'Sitio Web', 12);
INSERT INTO SAPW_Tipo VALUES(33, 'Cyber Links', 4);
INSERT INTO SAPW_Tipo VALUES(34, 'High Tech Information', 4);
INSERT INTO SAPW_Tipo VALUES(35, 'Privado', 13);
INSERT INTO SAPW_Tipo VALUES(36, 'Público', 13);
INSERT INTO SAPW_Tipo VALUES(37, 'Cédula', 14);
INSERT INTO SAPW_Tipo VALUES(38, 'Pasaporte', 14);
INSERT INTO SAPW_Tipo VALUES(39, 'Otros...', 5);
```


--Opciones del Portal del PROTCOM Santa Elena

```
INSERT INTO sapw_opcion_web VALUES (5, 0, 'Varios', 'ConstruccionI.php',
'ConstruccionI.php', 'ConstruccionI.php', 'ConstruccionI.php', 0, 'S', '<< Información no
disponible >>', '<< Información no disponible >>', 0, 5, 1);
INSERT INTO sapw_opcion_web VALUES (83, 0, 'Capacitacion', 'CapacitacionI.php',
'CapacitacionI.php', 'CapacitacionI.php', 'CapacitacionI.php', 0, 'V', '<< Información no
disponible >>', '<< Información no disponible >>', 0, 83, 1);
INSERT INTO sapw_opcion_web VALUES (53, 4, 'Perfil Docente',
'ConstruccionI.php', 'ConstruccionI.php', 'PerfilDocenteEducationI.php',
'ConstruccionI.php', 7, 'N', '<< Información no disponible >>', '<< Información no
disponible >>', 0, 53, 1);
INSERT INTO sapw_opcion_web VALUES (54, 2, 'Reservación de Laboratorios',
'ConstruccionI.php', 'ConstruccionI.php', 'Reservacion_Laboratorio_DC_I.php',
'ConstruccionI.php', 10, 'N', '<< Información no disponible >>', '<< Información no
disponible >>', 0, 54, 1);
INSERT INTO sapw_opcion_web VALUES (50, 4, 'Coordinación y Tutorías',
'ConstruccionI.php', 'ConstruccionI.php', 'CoordinadoresEducationI.php',
'ConstruccionI.php', 8, 'N', '<< Información no disponible >>', '<< Información no
disponible >>', 0, 50, 1);
INSERT INTO sapw_opcion_web VALUES (48, 4, 'Calificaciones',
'ConstruccionI.php', 'ConstruccionI.php', 'CalificacionEducationI.php',
'ConstruccionI.php', 10, 'N', '<< Información no disponible >>', '<< Información no
disponible >>', 0, 48, 1);
INSERT INTO sapw_opcion_web VALUES (49, 4, 'Historial Académico',
'ConstruccionI.php', 'ConstruccionI.php', 'HistorialEducationI.php', 'ConstruccionI.php',
15, 'N', '<< Información no disponible >>', '<< Información no disponible >>', 0, 49, 1);
INSERT INTO sapw_opcion_web VALUES (60, 5, 'Ayuda en Línea',
'AyudaenLineaIP.php', 'AyudaenLineaIP.php', 'ConstruccionI.php', 'ConstruccionI.php',
22, 'V', '<< Información no disponible >>', '<< Información no disponible >>', 0, 60, 1);
INSERT INTO sapw_opcion_web VALUES (111, 2, 'Protcom Chat',
'../ChatRoom/chatroom.html', '../ChatRoom/chatroom.html', 'ProtcomChatID.php',
'ProtcomChatID.php', 20, 'N', 'Interactuar en línea con los docentes', 'Ahora las
tecnologías emergentes coadyuvarán en el proceso de enseñanza aprendizaje
permitiendo, vía mensajería instantánea, interacción entre nuestros educadores y los
beneficiarios de nuestros productos educativos.', 0, 108, 1);
INSERT INTO sapw_opcion_web VALUES (108, 3, 'Cambiar Contraseña',
'Negocios_ComunidadI.php', 'Negocios_ComunidadI.php', 'ContrasenaNI.php',
'ContrasenaNI.php', 13, 'N', 'Permite cambiar la contraseña de los usuarios', 'Permite
cambiar la contraseña de los usuarios.', 0, 105, 1);
INSERT INTO sapw_opcion_web VALUES (78, 5, 'News & Publications',
'PaperPublicationI.php', 'PaperPublicationI.php', 'ConstruccionI.php',
'ConstruccionI.php', 89, 'V', '<< Información no disponible >>', '<< Información no
disponible >>', 0, 78, 1);
INSERT INTO sapw_opcion_web VALUES (99, 27, 'E-Education',
'Docentes_CampusI.php', 'ConstruccionI.php', 'NegociosEducation_DC_IP.php',
'ConstruccionI.php', 44, 'S', '<< Información no disponible >>', 'Trabajar en la
aplicación y desarrollo de estrategias de formación y aprendizaje a distancia a través del
uso de las nuevas tecnologías de la información y la comunicación.', 0, 99, 1);
INSERT INTO sapw_opcion_web VALUES (16, 1, 'Buzón virtual',
'Alumnos_PadresI.php', 'Alumnos_PadresI.php', 'BuzonVirtualI.php',
```

'BuzonVirtualI.php', 72, 'N', 'Buzón de opiniones on-line del PROTCOM Campus Santa Elena.', 'Este buzón es un ente recolector de las opiniones de los educandos y demás beneficiarios de servicios del PROTCOM Campus Santa Elena.', 0, 16, 1);

INSERT INTO sapw_opcion_web VALUES (84, 83, 'Seminarios de Capacitación', 'SeminarioCapacita.php', 'CapacitacionI.php', 'ConstruccionI.php', 'CapacitacionI.php', 88, 'V', 'La capacitación constante es parte del nuevo siglo.', 'Entrenar al participante en el manejo de los programas de mayor uso, ampliando sus capacidades de trabajo al aprender los principales software que trabajan bajo Windows.', 0, 84, 1);

INSERT INTO sapw_opcion_web VALUES (11, 1, 'Reservación de Laboratorios', 'Alumnos_PadresI.php', 'Alumnos_PadresI.php', 'ReservacionLaboratorioI.php', 'ReservacionLaboratorioI.php', 14, 'N', 'Reservar tiempo de máquina para desarrollo de proyecto.', 'La reservación de laboratorios está sujeta a las políticas proveídas por el Jefe de Redes contempladas en la distribución de recursos de laboratorios, a beneficios de los usuarios, sean éstos estudiantes o público en general.

La reservación deberá hacerse con la antelación correspondiente siguiendo los trámites del caso. Toda reservación deberá ser confirmada, al menos, tres horas antes de su ejecución.', 0, 11, 1);

INSERT INTO sapw_opcion_web VALUES (74, 5, 'Protcom Chat', '../ChatRoom/chatroom.html', '../ChatRoom/chatroom.html', 'ConstruccionI.php', 'ConstruccionI.php', 153, 'V', '<< Información no disponible >>', '<< Información no disponible >>', 0, 74, 1);

INSERT INTO sapw_opcion_web VALUES (6, 1, 'Rendimiento Académico', 'Alumnos_PadresI.php', 'Alumnos_PadresI.php', 'RendimientoAcademicoI.php', 'ConstruccionI.php', 32, 'N', 'Revisar notas de materias en curso.', 'Revisa las calificaciones de las materias registradas en el semestre o ciclo modular actual y planificar el tiempo de estudio para superarlas. Estas calificaciones son parciales o finales.', 0, 6, 1);

INSERT INTO sapw_opcion_web VALUES (129, 83, 'Proyectos de Capacitación', 'CapacitacionI.php', 'CapacitacionI.php', 'ProyectoCapacitaIP.php', 'CapacitacionI.php', 166, 'V', 'En este apartado se mostrará como se ha capacitado a un determinado grupo de personas, a través de convenios y/o proyectos.', '
CURSO DE COMPUTACIÓN PARA NIÑOS: Se refieren al dictado de clases de computación para los alumnos de las escuelas de la Península, entre los establecimientos educativos están:
- ESCUELA PARTICULAR "JUAN JACOBO ROUSSEAU"
- ESCUELA FISCAL DE NIÑAS "TEODORO WOLF"

CAPACITACIÓN PARA LOS PROFESORES DE INFORMÁTICA DE LOS COLEGIOS DE LA PENÍNSULA: A través de un convenio con el Programa de Desarrollo para la Península de Santa Elena (PDPSE), se impartió durante 4 meses, seminarios de Informática para los profesores que dictan materias en la especialización de Computación de los colegios peninsulares.

PROYECTO DE CAPACITACIÓN PARA LOS PROFESORES DE LAS ESPECIALIZACIONES DE COMERCIO Y ADMINISTRACIÓN, COMERCIALIZACIÓN, COMPUTACIÓN Y SECRETARIADO DE LOS COLEGIOS FISCALES DE LA PENÍNSULA: también con el apoyo del PDPSE. Proyecto que se compone de 2 etapas y que tiene como objetivo, mejorar la calidad de enseñanza educativa en los planteles fiscales del medio. En la primera etapa, se imparten seminarios de Estrategias Metodológicas, Trabajo en Equipo, Relaciones Humanas e Informática. Los colegios que participaron fueron:
- COLEGIO FISCAL TÉCNICO "ANCÓN"

- COLEGIO FISCAL TÉCNICO "MUEY"

 - COLEGIO FISCAL "DR. LUIS CÉLLERI AVILÉS"

 - COLEGIO FISCAL "PENÍNSULA DE SANTA ELENA"

 - COLEGIO FISCAL "PEDRO FRANCO DÁVILA"

 - COLEGIO A DISTANCIA "IRFEYAL"

Actualmente este proyecto se encuentra en la fase de estudio de su segunda etapa, que consiste en capacitar a los profesores en cada área de especialización. Con la experiencia de la primera etapa, y ante el interés mostrado, se tiene planificado desarrollar este proyecto, con un nuevo grupo de colegios.', 0, 86, 1);
 INSERT INTO sapw_opcion_web VALUES (65, 5, 'Programoteca',
 'ProgramotecaPubl.php', 'ConstruccionI.php', 'ConstruccionI.php', 'ConstruccionI.php',
 21, 'V', '<< Información no disponible >>', '<< Información no disponible >>', 0, 65, 1);
 INSERT INTO sapw_opcion_web VALUES (131, 27, 'Proyectos de Capacitación',
 'Docentes_CampusI.php', 'CapacitacionI.php', 'ProyectoCapacita_DC_IP.php',
 'CapacitacionI.php', 101, 'S', '<< Información no disponible >>', '
CURSO
 DE COMPUTACIÓN PARA NIÑOS: Se refieren al dictado de clases de
 computación para los alumnos de las escuelas de la Península, entre los
 establecimientos educativos están:

 - ESCUELA PARTICULAR "JUAN JACOBO ROUSSEAU"

 - ESCUELA FISCAL DE NIÑAS "TEODORO WOLF"

CAPACITACIÓN PARA LOS PROFESORES DE INFORMÁTICA
 DE LOS COLEGIOS DE LA PENÍNSULA: A través de un convenio con el
 Programa de Desarrollo para la Península de Santa Elena (PDPSE), se impartió durante
 4 meses, seminarios de Informática para los profesores que dictan materias en la
 especialización de Computación de los colegios peninsulares.

PROYECTO DE CAPACITACIÓN PARA LOS PROFESORES DE
 LAS ESPECIALIZACIONES DE COMERCIO Y ADMINISTRACIÓN,
 COMERCIALIZACIÓN, COMPUTACIÓN Y SECRETARIADO DE LOS
 COLEGIOS FISCALES DE LA PENÍNSULA: también con el apoyo del
 PDPSE. Proyecto que se compone de 2 etapas y que tiene como objetivo, mejorar la
 calidad de enseñanza educativa en los planteles fiscales del medio. En la primera etapa,
 se imparten seminarios de Estrategias Metodológicas, Trabajo en Equipo, Relaciones
 Humanas e Informática. Los colegios que participaron fueron:

 - COLEGIO FISCAL TÉCNICO "ANCÓN"

 - COLEGIO FISCAL TÉCNICO "MUEY"

 - COLEGIO FISCAL "DR. LUIS CÉLLERI AVILÉS"

 - COLEGIO FISCAL "PENÍNSULA DE SANTA ELENA"

 - COLEGIO FISCAL "PEDRO FRANCO DÁVILA"

 - COLEGIO A DISTANCIA "IRFEYAL"

Actualmente este proyecto se encuentra en la fase de estudio de su segunda etapa, que consiste en capacitar a los profesores en cada área de especialización. Con la experiencia de la primera etapa, y ante el interés mostrado, se tiene planificado desarrollar este proyecto, con un nuevo grupo de colegios.', 0, 82, 1);
 INSERT INTO sapw_opcion_web VALUES (126, 27, 'Carreras Universitarias',
 'PerfilCarreraI.php?select=1', 'ConstruccionI.php', 'Carrera_DC_IP.php',
 'ConstruccionI.php', 126, 'S', '<< Información no disponible >>', '
En el
 PROTCOM Campus Santa Elena se ofrecen las siguientes carreras
 modulares:

 - Secretariado Ejecutivo en Sistemas de Información

 - Programación de Sistemas


```

<strong>- </strong>Análisis de Soporte de Microcomputadores', 0, 108, 1);
INSERT INTO sapw_opcion_web VALUES (120, 5, 'Trámites Académicos',
'Docentes_CampusI.php', 'Docentes_CampusI.php', 'Docentes_CampusI.php',
'ConstruccionI.php', 173, 'V', 'Información de los pasos a seguir para realizar cambios
de carreras, convalidaciones de materias (ESPOL/otras universidades), certificados
varios, lineamientos que debes seguir para graduarte, datos que debes conocer antes de
registrarte, reglas de asistencia, reglas de evaluación en PROTCOM Campus Santa
Elena.', 'Información de los pasos a seguir para realizar cambios de carreras,
convalidaciones de materias (ESPOL/otras universidades), certificados varios,
lineamientos que debes seguir para graduarte, datos que debes conocer antes de
registrarte, reglas de asistencia, reglas de evaluación en PROTCOM Campus Santa
Elena.', 0, 19, 1);
INSERT INTO sapw_opcion_web VALUES (77, 5, 'Portafolios On-Line',
'ConstruccionI.php', 'ConstruccionI.php', 'ConstruccionI.php', 'ConstruccionI.php', 44,
'V', '<< Información no disponible >>', '<< Información no disponible >>', 0, 77, 1);
INSERT INTO sapw_opcion_web VALUES (3, 0, 'Negocios & Comunidad',
'Negocios_ComunidadI.php', 'Negocios_ComunidadN.php',
'Negocios_ComunidadI.php', 'Negocios_ComunidadN.php', 159, 'N', '<< Información
no disponible >>', '<< Información no disponible >>', 0, 3, 1);
INSERT INTO sapw_opcion_web VALUES (112, 3, 'Protcom Chat',
'../ChatRoom/chatroom.html', '../ChatRoom/chatroom.html', 'ProtcomChatIN.php',
'ProtcomChatIN.php', 17, 'N', 'Interactuar en línea con los docentes.', 'Ahora las
tecnologías emergentes coadyuvarán en el proceso de enseñanza aprendizaje
permitiendo, vía mensajería instantánea, interacción entre nuestros educadores y los
beneficiarios de nuestros productos educativos.', 0, 109, 1);
INSERT INTO sapw_opcion_web VALUES (8, 1, 'Historial Académico',
'Alumnos_PadresI.php', 'Alumnos_PadresI.php', 'HistorialAcademicoI.php',
'ConstruccionI.php', 60, 'N', 'Recuperar rendimiento académico de semestres o ciclos
anteriores.', 'Obtener la historia del rendimiento académico estudiantil, del alumno
consultado.', 0, 8, 1);
INSERT INTO sapw_opcion_web VALUES (106, 1, 'Cambiar Contraseña',
'Alumnos_PadresI.php', 'Alumnos_PadresI.php', 'ContrasenaAI.php',
'ContrasenaAI.php', 44, 'N', 'Permite Cambiar la Contraseña de los Usuarios.', 'Permite
Cambiar la Contraseña de los Usuarios.', 0, 103, 1);
INSERT INTO sapw_opcion_web VALUES (100, 9, 'Políticas de Curso',
'Alumnos_PadresI.php', 'Alumnos_PadresI.php', 'PoliticacursoI.php',
'CronogramaEstudioI.php', 13, 'N', 'Breve descripción de la malla curricular.', 'Consulta
en pantalla de los objetivos específicos, generales y de término de las materias
componentes de la malla curricular de la especialidad o carrera consultada.', 0, 36, 1);
INSERT INTO sapw_opcion_web VALUES (101, 9, 'Capítulos de Estudio',
'Alumnos_PadresI.php', 'Alumnos_PadresI.php', 'CapitulosEstudioI.php',
'CronogramaEstudioI.php', 11, 'N', 'Breve descripción de la malla curricular.', 'Consulta
en pantalla de los objetivos específicos, generales y de término de las materias
componentes de la malla curricular de la especialidad o carrera consultada.', 0, 34, 1);
INSERT INTO sapw_opcion_web VALUES (102, 9, 'Deberes', 'Alumnos_PadresI.php',
'Alumnos_PadresI.php', 'DeberesI.php', 'CronogramaEstudioI.php', 9, 'N', 'Breve
descripción de la malla curricular.', 'Consulta en pantalla de los objetivos específicos,
generales y de término de las materias componentes de la malla curricular de la
especialidad o carrera consultada.', 0, 31, 1);

```

INSERT INTO sapw_opcion_web VALUES (9, 1, 'Guía de Estudio', 'Alumnos_PadresI.php', 'Alumnos_PadresI.php', 'CronogramaEstudioI.php', 'CronogramaEstudioI.php', 33, 'N', 'Breve descripción de la malla curricular.', 'Consulta en pantalla de los objetivos específicos, generales y de término de las materias componentes de la malla curricular de la especialidad o carrera consultada.', 0, 9, 1);

INSERT INTO sapw_opcion_web VALUES (7, 1, 'Ranking', 'Alumnos_PadresI.php', 'Alumnos_PadresI.php', 'RankingI.php', 'ConstruccionI.php', 23, 'N', 'Chequear aprovechamiento estudiantil.', 'Chequear posición estudiantil a nivel de todos los estudiantes activos registrados en todas las carreras del PROTCOM Campus Santa Elena, tanto a nivel de carreras regulares o modulares. Esta consulta generará un listado en pantalla de los mejores 20 alumnos y la posición del estudiante consultado.', 0, 7, 1);

INSERT INTO sapw_opcion_web VALUES (29, 2, 'Programoteca', 'Docentes_CampusI.php', 'Docentes_CampusI.php', 'Programoteca_DC_I.php', 'Programoteca_DC_I.php', 34, 'N', 'Productos desarrollados por los estudiantes del PROTCOM Campus Santa Elena.', 'La Programoteca es una base digital que contienen todos los proyectos desarrollados por alumnos de todos los niveles de nuestra unidad.', 0, 29, 1);

INSERT INTO sapw_opcion_web VALUES (23, 22, 'Calificaciones', 'Docentes_CampusI.php', 'Docentes_CampusI.php', 'CalificacionesI.php', 'CalificacionesI.php', 8, 'N', 'Verificar las notas que ha obtenido el estudiante en las materias en cursos.', 'Ahora usted tendrá un seguimiento más exhaustivo de las notas que un estudiante a nivel de todas las carreras del PROTCOM Campus Santa Elena.', 0, 23, 1);

INSERT INTO sapw_opcion_web VALUES (104, 22, 'Deberes', 'Docentes_CampusI.php', 'Docentes_CampusI.php', 'Deberes_DC_I.php', 'Deberes_DC_I.php', 8, 'N', 'Consultar el contenido de las materias que se dictan en cada una de las carreras del PROTCOM Campus Santa Elena.', 'Consulte el contenido de cada una de las materias que se dictan en cada una de las carreras del PROTCOM Campus Santa Elena.', 0, 104, 1);

INSERT INTO sapw_opcion_web VALUES (103, 22, 'Políticas de Curso', 'Docentes_CampusI.php', 'Docentes_CampusI.php', 'PoliticasyCurso_DC_I.php', 'PoliticasyCurso_DC_I.php', 7, 'N', 'Consultar el contenido de las materias que se dictan en cada una de las carreras del PROTCOM Campus Santa Elena.', 'Consulte el contenido de cada una de las materias que se dictan en cada una de las carreras del PROTCOM Campus Santa Elena.', 0, 103, 1);

INSERT INTO sapw_opcion_web VALUES (24, 22, 'Capítulos de Estudio', 'Docentes_CampusI.php', 'Docentes_CampusI.php', 'CapitulosEstudio_DC_I.php', 'CapitulosEstudio_DC_I.php', 7, 'N', 'Consultar el contenido de las materias que se dictan en cada una de las carreras del PROTCOM Campus Santa Elena.', 'Consulte el contenido de cada una de las materias que se dictan en cada una de las carreras del PROTCOM Campus Santa Elena.', 0, 24, 1);

INSERT INTO sapw_opcion_web VALUES (25, 22, 'Perfil Docente', 'Docentes_CampusI.php', 'Docentes_CampusI.php', 'PerfilDocenteI.php', 'PerfilDocenteI.php', 6, 'N', 'Consultar el curriculum de los docentes del PROTCOM Campus Santa Elena.', 'Ahora podrás ver el curriculum vitae de cada uno de los profesores que dictan clase en PROTCOM Campus Santa Elena en todas las carreras.', 0, 25, 1);

INSERT INTO sapw_opcion_web VALUES (22, 2, 'Web Académico', 'Docentes_CampusI.php', 'Docentes_CampusI.php', 'WebAcademicoI.php', 'WebAcademicoI.php', 36, 'N', 'Chequear aspectos académicos del PROTCOM Campus

Santa Elena.', 'La opción Web Académico, permite familiarizarse con los aspectos académico de nuestra unidad.', 0, 22, 1);

INSERT INTO sapw_opcion_web VALUES (107, 2, 'Cambiar Contraseña', 'Docentes_CampusI.php', 'Docentes_CampusI.php', 'ContrasenaDI.php', 'ContrasenaDI.php', 23, 'N', 'Permite Cambiar la Contraseña de los Usuarios.', 'Permite cambiar la contraseña de los usuarios.', 0, 104, 1);

INSERT INTO sapw_opcion_web VALUES (105, 2, 'Buzón virtual', 'Docentes_CampusI.php', 'Docentes_CampusI.php', 'BuzonVirtual_DC_I.php', 'BuzonVirtual_DC_I.php', 15, 'N', 'Buzón de opiniones on-line del PROTCOM Campus Santa Elena.', 'Este buzón es un ente recolector de las opiniones de los educandos y demás beneficiarios de servicios del PROTCOM Campus Santa Elena.', 0, 102, 1);

INSERT INTO sapw_opcion_web VALUES (28, 2, 'Backup On-line', 'Docentes_CampusI.php', 'Docentes_CampusI.php', 'BackupOnLine_DC_I.php', 'BackupOnLine_DC_I.php', 8, 'N', 'Respalidar archivos y trabajos de los educandos.', 'Los estudiantes registrados en las carreras, certificados y demás cursos al público, podrán previa autorización alojar sus trabajos temporalmente. El uso de este beneficio será normado por las políticas de manejo de recursos de hardware y software de la unidad.', 0, 28, 1);

INSERT INTO sapw_opcion_web VALUES (133, 2, 'Seminarios de Capacitación', 'SeminarioCapacita.php', 'ConstruccionI.php', 'CertificadoIP.php', 'ConstruccionI.php', 62, 'V', 'La Capacitación constante es parte de la Nueva Era.', 'La Capacitación constante es parte de la Nueva Era.', 0, 41, 1);

INSERT INTO sapw_opcion_web VALUES (12, 1, 'Financiero estudiantil', 'Alumnos_PadresI.php', 'Alumnos_PadresI.php', 'FinancieroAcademicoI.php', 'FinancieroAcademicoI.php', 23, 'N', 'Verificar deudas o saldos pendientes con el PROTCOM Campus Santa Elena o la institución.', 'Es importante que todo estudiante verifique si tiene deuda o saldo pendiente antes de los periodos de registros o para los trámites de graduación. Además el no adeudar ayudará a facilitar la tramitación interna de documentos o certificados.', 0, 12, 1);

INSERT INTO sapw_opcion_web VALUES (88, 86, 'Análisis de Soporte de Microcomputadores', 'PerfilCarreraI.php?select=1', 'CapacitacionI.php', 'PerfilCarreraI.php?select=1', 'PerfilCarreraI.php?select=1', 56, 'V', 'Dar al estudiante los conocimientos básicos y complementarios del análisis de sistemas orientado única y exclusivamente al campo de los Microcomputadores. El estudiante estará en capacidad de dirigir y manipular los recursos informáticos de las empresas.', 'Dar al estudiante los conocimientos básicos y complementarios del análisis de sistemas orientado única y exclusivamente al campo de los Microcomputadores. El estudiante estará en capacidad de dirigir y manipular los recursos informáticos de las empresas.', 0, 88, 1);

INSERT INTO sapw_opcion_web VALUES (2, 0, 'Docentes & Campus', 'Docentes_CampusI.php', 'Docentes_CampusN.php', 'Docentes_CampusI.php', 'Docentes_CampusN.php', 368, 'N', '<< Información no disponible >>', '<< Información no disponible >>', 0, 2, 1);

INSERT INTO sapw_opcion_web VALUES (132, 40, 'Proyectos de Capacitación', 'Negocios_ComunidadI.php', 'CapacitacionI.php', 'ProyectoCapacitaIP.php', 'CapacitacionI.php', 96, 'S', '<< Información no disponible >>', '
CURSO DE COMPUTACIÓN PARA NIÑOS: Se refieren al dictado de clases de computación para los alumnos de las escuelas de la Península, entre los establecimientos educativos están:

- ESCUELA PARTICULAR "JUAN JACOBO ROUSSEAU"

- ESCUELA FISCAL DE NIÑAS "TEODORO WOLF"

CAPACITACIÓN PARA LOS PROFESORES DE INFORMÁTICA DE LOS COLEGIOS DE LA PENÍNSULA: A través de un convenio con el Programa de Desarrollo para la Península de Santa Elena (PDPSE), se impartió durante 4 meses, seminarios de Informática para los profesores que dictan materias en la especialización de Computación de los colegios peninsulares.

PROYECTO DE CAPACITACIÓN PARA LOS PROFESORES DE LAS ESPECIALIZACIONES DE COMERCIO Y ADMINISTRACIÓN, COMERCIALIZACIÓN, COMPUTACIÓN Y SECRETARIADO DE LOS COLEGIOS FISCALES DE LA PENÍNSULA: también con el apoyo del PDPSE. Proyecto que se compone de 2 etapas y que tiene como objetivo, mejorar la calidad de enseñanza educativa en los planteles fiscales del medio. En la primera etapa, se imparten seminarios de Estrategias Metodológicas, Trabajo en Equipo, Relaciones Humanas e Informática. Los colegios que participaron fueron:

- COLEGIO FISCAL TÉCNICO "ANCÓN"

- COLEGIO FISCAL TÉCNICO "MUEY"

- COLEGIO FISCAL "DR. LUIS CÉLLERI AVILÉS"

- COLEGIO FISCAL "PENÍNSULA DE SANTA ELENA"

- COLEGIO FISCAL "PEDRO FRANCO DÁVILA"

- COLEGIO A DISTANCIA "IRFEYAL"

Actualmente este proyecto se encuentra en la fase de estudio de su segunda etapa, que consiste en capacitar a los profesores en cada área de especialización. Con la experiencia de la primera etapa, y ante el interés mostrado, se tiene planificado desarrollar este proyecto, con un nuevo grupo de colegios.', 0, 82, 1);

INSERT INTO sapw_opcion_web VALUES (44, 40, 'E-Education', 'Negocios_ComunidadI.php', 'ConstruccionI.php', 'NegociosEducationIP.php', 'ConstruccionI.php', 55, 'S', '<< Información no disponible >>', 'Trabajar en la aplicación y desarrollo de estrategias de formación y aprendizaje a distancia a través del uso de las nuevas tecnologías de la información y la comunicación.', 0, 44, 1);
INSERT INTO sapw_opcion_web VALUES (127, 40, 'Carreras Universitarias', 'PerfilCarreral.php?select=1', 'ConstruccionI.php', 'CarreraIP.php', 'ConstruccionI.php', 143, 'S', '<< Información no disponible >>', '
En el PROTCOM Campus Santa Elena se ofrecen las siguientes carreras modulares:

- Secretariado Ejecutivo en Sistemas de Información

- Programación de Sistemas

- Análisis de Soporte de Microcomputadores', 0, 108, 1);

INSERT INTO sapw_opcion_web VALUES (63, 5, 'Pasantías Virtuales', 'ConstruccionI.php', 'ConstruccionI.php', 'ConstruccionI.php', 'ConstruccionI.php', 24, 'V', '<< Información no disponible >>', '<< Información no disponible >>', 0, 63, 1);
INSERT INTO sapw_opcion_web VALUES (135, 2, 'Agenda Cultural del PROTCOM Santa Elena', 'EventosI.php', 'EventosI.php', 'EventosI.php', 'EventosI.php', 55, 'E', 'Eventos realizados en el Programa de Tecnología en Computación y Diseño Gráfico (PROTCOM Campus Santa Elena).', 'Eventos realizados en el Programa de Tecnología en Computación y Diseño Gráfico (PROTCOM Campus Santa Elena).', 0, 237, 1);

INSERT INTO sapw_opcion_web VALUES (52, 4, 'Plana Docente', 'E_EducationI.php', 'ConstruccionI.php', 'PlanaDocenteEducationI.php', 'ConstruccionI.php', 101, 'S', 'E-Education pone a su disposición su mejor personal docente.', 'E-Education pone a su disposición su mejor personal docente.', 0, 52, 1);
INSERT INTO sapw_opcion_web VALUES (61, 5, 'Links de interés', 'CyberLinksI.php', 'CyberLinksI.php', 'CyberLinksI.php', 'CyberLinksI.php', 71, 'V', '<< Información no disponible >>', '<< Información no disponible >>', 0, 61, 1);

INSERT INTO sapw_opcion_web VALUES (62, 5, 'High Tech Information', 'HighTechInformationI.php', 'HighTechInformationI.php', 'HighTechInformationI.php', 'HighTechInformationI.php', 49, 'V', '<< Información no disponible >>', '<< Información no disponible >>', 0, 62, 1);

INSERT INTO sapw_opcion_web VALUES (15, 1, 'Backup On-line', 'Alumnos_PadresI.php', 'Alumnos_PadresI.php', 'BackupOnLineI.php', 'BackupOnLineI.php', 29, 'N', 'Respaldo archivos y trabajos de los educandos.', 'Los estudiantes registrados en las carreras, certificados y demás cursos al público, podrán previa autorización alojar sus trabajos temporalmente. El uso de este beneficio será normado por las políticas de manejo de recursos de hardware y software de la unidad.', 0, 15, 1);

INSERT INTO sapw_opcion_web VALUES (18, 1, 'Perfil estudiantil', 'Alumnos_PadresI.php', 'Alumnos_PadresI.php', 'Perfil_AcademicoI.php', 'Perfil_AcademicoI.php', 26, 'N', 'Niveles de Acceso al Portal por perfil.', 'El visitante podrá a través login y password acceder a los portafolios de servicios del PROTCOM Campus Santa Elena, los cuales están clasificados en base a las soluciones integradas provistas.', 0, 18, 1);

INSERT INTO sapw_opcion_web VALUES (32, 2, 'Pasantías y clases compartidas', 'Docentes_CampusI.php', 'Docentes_CampusI.php', 'PasantiasClases_DC_I.php', 'PasantiasClases_DC_I.php', 115, 'S', 'Registro de pasantías a nivel de docentes y educandos', 'Seguidamente tiene a disposición un formulario para que registre sus datos, como educando o educador y en base a su perfil y demanda, califique para labores docentes o de ayudantía compartidas en el PROTCOM Campus Santa Elena. Esto coadyuvará en el proceso de enseñanza-aprendizaje interinstitucional de nuestra unidad con sus similares en otras instituciones educativas.', 0, 32, 1);

INSERT INTO sapw_opcion_web VALUES (45, 3, 'Soporte y Outsourcing de servicios', 'Negocios_ComunidadI.php', 'ConstruccionI.php', 'SoporteOutsourI.php', 'ConstruccionI.php', 63, 'S', 'Ofrecemos soluciones óptimas que son motivo de permanentes alianzas estratégicas, haciéndonos cargo de todos los procesos que no forman parte de la competencia principal del negocio de su organización.', 'Ofrecemos soluciones óptimas que son motivo de permanentes alianzas estratégicas, haciéndonos cargo de todos los procesos que no forman parte de la competencia principal del negocio de su organización.', 0, 45, 1);

INSERT INTO sapw_opcion_web VALUES (47, 4, 'Portafolio de Servicios Académicos on-line', 'E_EducationI.php', 'ConstruccionI.php', 'ServicioPortafolioEducationI.php', 'ConstruccionI.php', 141, 'S', 'El Programa de Tecnología en Computación (PROTCOM) Campus Santa Elena ha desarrollado un programa de educación a distancia, de ese modo podrá capacitarse en un horario de tiempo más flexible que si estuviera en un curso tradicional y disfrutará de la garantía de tener una educación acorde con los tiempos actuales.', 'El PROTCOM Campus Santa Elena ve en las tecnologías de la comunicación y la información instrumentos efectivos para brindar sus servicios y fortalecer la cooperación académica, la investigación y el desarrollo tecnológico.', 0, 47, 1);

INSERT INTO sapw_opcion_web VALUES (109, 4, 'Cambiar contraseña', 'E_EducationI.php', 'E_EducationI.php', 'ContrasenaEI.php', 'ContrasenaEI.php', 6, 'N', 'Permite cambiar la contraseña de los usuarios.', 'Permite cambiar la contraseña de los usuarios.', 0, 109, 1);

INSERT INTO sapw_opcion_web VALUES (51, 4, 'Web Content', 'E_EducationI.php', 'ConstruccionI.php', 'WebContentEducationI.php', 'ConstruccionI.php', 70, 'S', 'Podrás

ver el contenido de todas las opciones que tenemos para tí.', 'Podrás ver el contenido de todas las opciones que tenemos para tí.', 0, 51, 1);

INSERT INTO sapw_opcion_web VALUES (113, 4, 'Protcom Chat',
'../ChatRoom/chatroom.html', '../ChatRoom/chatroom.html', 'ProtcomChatIE.php',
'ProtcomChatIE.php', 15, 'N', 'Interactuar en línea con los docentes.', 'Ahora las tecnologías emergentes coadyuvan en el proceso de enseñanza aprendizaje permitiendo, vía mensajería instantánea, interacción entre nuestros educadores y los beneficiarios de nuestros productos educativos.', 0, 110, 1);

INSERT INTO sapw_opcion_web VALUES (46, 4, 'Nuevo Registro',
'E_EducationI.php', 'ConstruccionI.php', 'RegistroEducationI.php', 'ConstruccionI.php',
80, 'S', 'Vamos regístrate y obtendrás muchos beneficios.', 'Desde esta opción podrás registrarte y así formar parte de la nueva era en la educación.', 0, 46, 1);

INSERT INTO sapw_opcion_web VALUES (82, 10, 'E-Education',
'Alumnos_PadresI.php', 'ConstruccionI.php', 'NegociosEducation_AP_IP.php',
'ConstruccionI.php', 81, 'S', '<< Información no disponible >>', 'Trabajar en la aplicación y desarrollo de estrategias de formación y aprendizaje a distancia a través del uso de las nuevas tecnologías de la información y la comunicación.', 0, 81, 1);

INSERT INTO sapw_opcion_web VALUES (1, 0, 'Alumnos & Padres',
'Alumnos_PadresI.php', 'Alumnos_PadresI.php', 'Alumnos_PadresI.php',
'Alumnos_PadresN.php', 451, 'S', '<< Información no disponible >>', '<< Información no disponible >>', 0, 1, 1);

INSERT INTO sapw_opcion_web VALUES (89, 86, 'Secretariado Ejecutivo en Sistemas de Información', 'PerfilCarreraI.php?select=3', 'CapacitacionI.php',
'CapacitacionI.php', 'CapacitacionI.php', 46, 'V', 'La Escuela Superior Politécnica del Litoral a través del Programa de Tecnología en Computación y Diseño Gráfico (PROTCOM), presenta al público en general la carrea modular de Secretariado Ejecutivo en Sistemas de Información, como un apoyo al sector empresarial y en beneficio del país. Esta carrera tiene el objetivo de entregar al mercado profesionales en el campo administrativo empresarial y áreas afines a fin de convertirlos en eficientes asistentes en el apoyo de toma de decisiones, utilizando la informática como un medio efectivo de incrementar su productividad.', 'La Escuela Superior Politécnica del Litoral a través del Programa de Tecnología en Computación y Diseño Gráfico (PROTCOM), presenta al público en general la carrea modular de Secretariado Ejecutivo en Sistemas de Información, como un apoyo al sector empresarial y en beneficio del país. Esta carrera tiene el objetivo de entregar al mercado profesionales en el campo administrativo empresarial y áreas afines a fin de convertirlos en eficientes asistentes en el apoyo de toma de decisiones, utilizando la informática como un medio efectivo de incrementar su productividad.', 0, 89, 1);

INSERT INTO sapw_opcion_web VALUES (87, 86, 'Programación de Sistemas',
'PerfilCarreraI.php?select=2', 'CapacitacionI.php', 'PerfilCarreraI.php?select=2',
'PerfilCarreraI.php?select=2', 145, 'V', 'Dar al estudiante las habilidades fundamentales en el procesamiento de datos, aplicando técnicas de programación actualizadas.', 'Dar al estudiante las habilidades fundamentales en el procesamiento de datos, aplicando técnicas de programación actualizadas.', 0, 87, 1);

INSERT INTO sapw_opcion_web VALUES (86, 83, 'Carreras Técnicas',
'CapacitacionI.php', 'CapacitacionI.php', 'CapacitacionI.php', 'CapacitacionI.php', 263,
'V', '<< Información no disponible >>', '<< Información no disponible >>', 0, 85, 1);

INSERT INTO sapw_opcion_web VALUES (96, 27, 'Seminarios de Capacitación',
'SeminarioCapacita.php', 'ConstruccionI.php', 'Certificado_DC_IP.php',
'ConstruccionI.php', 107, 'S', 'Entrenar al participante en el manejo de los programas de

mayor uso, ampliando sus capacidades de trabajo al aprender los principales software que trabajan bajo Windows.'

Principalmente al personal del sector productivo que desea mantener una constante preparación. Los constantes cambios a nivel informático nos obligan a mantenernos en un continuo entrenamiento, el cual hace posible alcanzar las metas propuestas en calidad y productividad. Por lo que PROTCOM Campus Santa Elena mantiene abiertos durante todo el año los Seminarios de Capacitación.

También están dirigidos al público en general, en especial a aquellas personas que no tienen ningún conocimiento en computación y que desean ingresar a un plan de capacitación progresiva en este amplio campo.

Modalidad de Estudio

Por cada seminario aprobado, el participante recibe su comprobante y además el crédito respectivo para que, tomando otros seminarios, en conjunto, le acreditan un certificado de capacitación en un área determinada, de mayor jerarquía y peso académico.

La elección de cada seminario es libre de acuerdo al gusto, a las posibilidades económicas y a la disponibilidad de horarios del participante.

Duración

Para acreditar un certificado de capacitación, el participante deberá completar un mínimo de horas requeridas.

Entregas de Certificados

Cada semestre, PROTCOM Campus Santa Elena realizará una ceremonia especial en donde se hará la entrega de los certificados de capacitación refrendado por la ESPOL que acredite haber aprobado los seminarios del programa correspondiente.

Se han otorgado hasta la presente fecha el Certificado de Capacitación de Asistente de Computación Nivel Avanzado a 11 promociones en este campus.', 0, 96, 1);

INSERT INTO sapw_opcion_web VALUES (19, 1, 'Protcom Chat',
'../ChatRoom/chatroom.html', '../ChatRoom/chatroom.html', 'ProtcomChatI.php',
'ProtcomChatI.php', 892, 'S', 'Interactúa en línea con otros usuarios y tendrás muchos amigos.', 'Ahora las tecnologías emergentes coadyuvará en el proceso de enseñanza-aprendizaje permitiendo, vía mensajería instantánea, interacción entre nuestros educadores y los beneficiarios de nuestros productos educativos.', 0, 20, 1);

INSERT INTO sapw_opcion_web VALUES (95, 5, 'Autores del Portal PROTCOM Santa Elena', 'AutoresI.php', 'ConstruccionI.php', 'ConstruccionI.php',
'ConstruccionI.php', 226, 'V', '<< Información no disponible >>', '<< Información no disponible >>', 0, 95, 1);

INSERT INTO sapw_opcion_web VALUES (125, 10, 'Carreras Universitarias',
'PerfilCarreraI.php?select=1', 'ConstruccionI.php', 'Carrera_AP_IP.php',
'ConstruccionI.php', 230, 'S', '<< Información no disponible >>',
PROTCOM Campus Santa Elena se ofrecen las siguientes carreras modulares:

- Secretariado Ejecutivo en Sistemas de Información

- Programación de Sistemas

- Análisis de Soporte de Microcomputadores', 0, 108, 1);

INSERT INTO sapw_opcion_web VALUES (130, 10, 'Proyectos de Capacitación',
'Alumnos_PadresI.php', 'ConstruccionI.php', 'ProyectoCapacita_AP_IP.php',
'ConstruccionI.php', 100, 'S', 'En este apartado se mostrará como se ha capacitado a un determinado grupo de personas, a través de convenios y/o proyectos.',

CURSO DE COMPUTACIÓN PARA NIÑOS: Se refieren al dictado de clases de computación para los alumnos de las escuelas de la Península, entre los establecimientos educativos están:

- ESCUELA PARTICULAR "JUAN JACOBO ROUSSEAU"

 - ESCUELA FISCAL DE NIÑAS "TEODORO WOLF"

CAPACITACIÓN PARA LOS PROFESORES DE INFORMÁTICA
 DE LOS COLEGIOS DE LA PENÍNSULA: A través de un convenio con el
 Programa de Desarrollo para la Península de Santa Elena (PDPSE) , se impartió durante
 4 meses, seminarios de Informática para los profesores que dictan materias en la
 especialización de Computación de los colegios peninsulares.

PROYECTO DE CAPACITACIÓN PARA LOS PROFESORES DE
 LAS ESPECIALIZACIONES DE COMERCIO Y ADMINISTRACIÓN,
 COMERCIALIZACIÓN, COMPUTACIÓN Y SECRETARIADO DE LOS
 COLEGIOS FISCALES DE LA PENÍNSULA: también con el apoyo del
 PDPSE. Proyecto que se compone de 2 etapas y que tiene como objetivo, mejorar la
 calidad de enseñanza educativa en los planteles fiscales del medio. En la primera etapa,
 se imparten seminarios de Estrategias Metodológicas, Trabajo en Equipo, Relaciones
 Humanas e Informática. Los colegios que participaron fueron:

 - COLEGIO FISCAL TÉCNICO "ANCÓN"

 - COLEGIO FISCAL TÉCNICO "MUEY"

 - COLEGIO FISCAL "DR. LUIS CÉLLERI AVILÉS"

 - COLEGIO FISCAL "PENÍNSULA DE SANTA ELENA"

 - COLEGIO FISCAL "PEDRO FRANCO DÁVILA"

 - COLEGIO A DISTANCIA "IRFEYAL"

Actualmente este proyecto se encuentra en la fase de estudio de su segunda
 etapa, que consiste en capacitar a los profesores en cada área de especialización. Con la
 experiencia de la primera etapa, y ante el interés mostrado, se tiene planificado
 desarrollar este proyecto, con un nuevo grupo de colegios.', 0, 82, 1);
 INSERT INTO sapw_opcion_web VALUES (13, 1, 'Pasantías y Ayudantías
 Académicas', 'Alumnos_PadresI.php', 'Alumnos_PadresI.php',
 'PasanAyudanAcademicasI.php', 'PasanAyudanAcademicasI.php', 232, 'S', 'Registro de
 pasantías a nivel de docentes y educandos.', 'Seguidamente tiene a disposición un
 formulario para que registre sus datos, como educando o educador y en base a su perfil
 y demanda, califique para labores docentes o de ayudantía compartidas en el
 PROTCOM Campus Santa Elena. Esto coadyuvará en el proceso de enseñanza-
 aprendizaje interinstitucional de nuestra unidad con sus similares en otras instituciones
 educativas.', 0, 13, 1);
 INSERT INTO sapw_opcion_web VALUES (118, 1, 'Foro de Discusión', 'ForumI.php',
 'ForumI.php', 'ForumI.php', 'ForumI.php', 157, 'S', 'Encuentra respuestas a tus dudas y
 problemas.', 'Los usuarios podrán plantear sus preguntas y encontrar respuestas sobre
 temas.', 0, 204, 1);
 INSERT INTO sapw_opcion_web VALUES (40, 3, 'Portafolio de Servicios
 Académicos', 'Negocios_ComunidadI.php', 'ConstruccionI.php',
 'Negocios_ComunidadI.php', 'ConstruccionI.php', 329, 'S', ' ', 'Brindamos servicios
 orientados a proporcionar ayudas a las diversas modalidades educativas que ofrece
 nuestra Universidad, así como facilitar la comunicación en aspectos académicos y de
 investigación con el uso de tecnología de cómputo de vanguardia.', 0, 40, 1);
 INSERT INTO sapw_opcion_web VALUES (41, 40, 'Seminarios de Capacitación',
 'SeminarioCapacita.php', 'ConstruccionI.php', 'CertificadoIP.php', 'ConstruccionI.php',
 93, 'S', '<< Información no disponible >>',
Dirigido a

 Principalmente al personal del sector productivo que desea mantener una constante
 preparación. Los constantes cambios a nivel informático nos obligan a mantenernos en
 un continuo entrenamiento, el cual hace posible alcanzar las metas propuestas en

calidad y productividad. Por lo que PROTCOM Campus Santa Elena mantiene abiertos durante todo el año los Seminarios de Capacitación.

También están dirigidos al público en general, en especial a aquellas personas que no tienen ningún conocimiento en computación y que desean ingresar a un plan de capacitación progresiva en este amplio campo.

Modalidad de Estudio

Por cada seminario aprobado, el participante recibe su comprobante y además el crédito respectivo para que, tomando otros seminarios, en conjunto, le acreditan un certificado de capacitación en un área determinada, de mayor jerarquía y peso académico.

La elección de cada seminario es libre de acuerdo al gusto, a las posibilidades económicas y a la disponibilidad de horarios del participante.

Duración

Para acreditar un certificado de capacitación, el participante deberá completar un mínimo de horas requeridas.

Entregas de Certificados

Cada semestre, PROTCOM Campus Santa Elena realizará una ceremonia especial en donde se hará la entrega de los certificados de capacitación refrendado por la ESPOL que acredite haber aprobado los seminarios del programa correspondiente.

Se han otorgado hasta la presente fecha el Certificado de Capacitación de Asistente de Computación Nivel Avanzado a 11 promociones en este campus.', 0, 41, 1);

INSERT INTO sapw_opcion_web VALUES (4, 0, 'E-education', 'E_EducationI.php', 'E_EducationN.php', 'E_EducationI.php', 'E_EducationN.php', 262, 'N', '<< Información no disponible >>', '<< Información no disponible >>', 0, 4, 1);

INSERT INTO sapw_opcion_web VALUES (79, 10, 'Seminarios de Capacitación', 'SeminarioCapacita.php', 'ConstruccionI.php', 'Certificado_AP_IP.php', 'ConstruccionI.php', 193, 'S', '<< Información no disponible >>', '
Dirigido a

Principalmente al personal del sector productivo que desea mantener una constante preparación. Los constantes cambios a nivel informático nos obligan a mantenernos en un continuo entrenamiento, el cual hace posible alcanzar las metas propuestas en calidad y productividad. Por lo que PROTCOM Campus Santa Elena mantiene abiertos durante todo el año los Seminarios de Capacitación.

También están dirigidos al público en general, en especial a aquellas personas que no tienen ningún conocimiento en computación y que desean ingresar a un plan de capacitación progresiva en este amplio campo.

Modalidad de Estudio

Por cada seminario aprobado, el participante recibe su comprobante y además el crédito respectivo para que, tomando otros seminarios, en conjunto, le acreditan un certificado de capacitación en un área determinada, de mayor jerarquía y peso académico.

La elección de cada seminario es libre de acuerdo al gusto, a las posibilidades económicas y a la disponibilidad de horarios del participante.

Duración

Para acreditar un certificado de capacitación, el participante deberá completar un mínimo de horas requeridas.

Entregas de Certificados

Cada semestre, PROTCOM Campus Santa Elena realizará una ceremonia especial en donde se hará la entrega de los certificados de capacitación refrendado por la ESPOL que acredite haber aprobado los seminarios del programa correspondiente.

Se han otorgado hasta la presente fecha el Certificado de Capacitación de Asistente de Computación Nivel Avanzado a 11 promociones en este campus.', 0, 79, 1);

INSERT INTO sapw_opcion_web VALUES (10, 1, 'Portafolio de Servicios Educativos', 'Alumnos_PadresI.php', 'Alumnos_PadresI.php', 'Alumnos_PadresI.php', 'ConstruccionI.php', 710, 'S', 'Seminarios, Certificados del PROTCOM Campus Santa Elena.', 'Dirigidos al público en general, personas que no tienen conocimientos en computación y que desean ingresar a un plan de capacitación progresiva en el campo informático. También están dirigidos al personal del sector empresarial, que desea mantener una constante preparación y actualización de conocimientos, útiles para beneficio del participante y de la empresa en general.

La base de estos certificados de capacitación se originan en los seminarios de computación que dicta el PROTCOM Campus Santa Elena de manera abierta al público. Una vez que el participante aprueba un seminario, nosotros lo guiamos para que tome otros seminarios relacionados, involucrándolo en un proceso de educación continua que lo capacitará en un área específica de la informática.

El participante está en la libertad de elegir el seminario que se adecúe a sus necesidades y posibilidades, aprobarlo y recibir su diploma respectivo. La aprobación de este seminario le acredita para la obtención futura de un "Certificado de Capacitación" de mayor jerarquía y peso académico.

La acreditación de un Certificado de Capacitación, se la obtiene cuando el estudiante haya completado un mínimo de horas de clases, que las puede obtener de un máximo de 6 seminarios con una duración de 30 horas de promedio. Un Certificado se lo puede obtener en un mínimo de 6 meses.', 0, 10, 1);

INSERT INTO sapw_opcion_web VALUES (17, 1, 'Programoteca', 'ProgramotecaPubI.php', 'Alumnos_PadresI.php', 'ProgramotecaI.php', 'ProgramotecaI.php', 477, 'S', 'Productos desarrollados por usuarios innovadores como tú.', 'La Programoteca es una base digital que contiene todos los proyectos desarrollados por alumnos de todos los niveles de nuestra unidad.', 0, 17, 1);

INSERT INTO sapw_opcion_web VALUES (122, 119, 'Lineamientos de Graduación', 'Docentes_CampusI.php', 'ConstruccionI.php', 'ConstruccionI.php', 'ConstruccionI.php', 126, 'S', '<< Información no disponible >>', 'Encontrarás todos los requisitos para convalidar tus materias ya sea que desees cambiarte de carrera o cambiarte de Universidad. Además te enterarás de los lineamientos que debes seguir para graduarte en PROTCOM Campus Santa Elena.

Descargar Lineamientos de Graduación', 0, 72, 1);

INSERT INTO sapw_opcion_web VALUES (123, 119, 'Informativo de Registro', 'Docentes_CampusI.php', 'ConstruccionI.php', 'ConstruccionI.php', 'ConstruccionI.php', 111, 'S', '<< Información no disponible >>', 'Datos que debes conocer antes de registrarte, reglas de asistencia, de evaluación. También sabrás como obtener becas dentro de PROTCOM Campus Santa Elena.

 Descargar Informativo de Registro', 0, 76, 1);

INSERT INTO sapw_opcion_web VALUES (124, 119, 'Emisión de Certificados', 'Docentes_CampusI.php', 'ConstruccionI.php', 'ConstruccionI.php', 'ConstruccionI.php', 108, 'S', '<< Información no disponible >>', 'Conocerás a donde debes recurrir para obtener tu certificado de egresado, asistencia, conducta o materias aprobadas.

 Descargar Emisión de Certificados', 0, 71, 1);

INSERT INTO sapw_opcion_web VALUES (121, 119, 'Consejería Académica', 'Docentes_CampusI.php', 'ConstruccionI.php', 'ConstruccionI.php', 'ConstruccionI.php', 104, 'S', 'Podrás encontrar información de qué es una Consejería Académica y cuál es su función?, quién es tu Consejera Académica asignada?, qué pasos debes seguir para registrarte?, y más', 'Podrás encontrar información de qué es una Consejería Académica y cuál es su función?, quién es tu Consejera Académica asignada?, qué pasos debes seguir para registrarte?, y más

 Descargar Consejería Académica', 0, 80, 1);

INSERT INTO sapw_opcion_web VALUES (119, 2, 'Trámites Académicos', 'Docentes_CampusI.php', 'Docentes_CampusI.php', 'Docentes_CampusI.php', 'ConstruccionI.php', 309, 'S', 'Información de los pasos a seguir para realizar cambios de carreras, convalidaciones de materias (ESPOL/otras universidades), certificados varios, lineamientos que debes seguir para graduarte, datos que debes conocer antes de registrarte, reglas de asistencia, reglas de evaluación en PROTCOM Campus Santa Elena.', 'Información de los pasos a seguir para realizar cambios de carreras, convalidaciones de materias (ESPOL/otras universidades), certificados varios, lineamientos que debes seguir para graduarte, datos que debes conocer antes de registrarte, reglas de asistencia, reglas de evaluación en PROTCOM Campus Santa Elena.', 0, 75, 1);

INSERT INTO sapw_opcion_web VALUES (26, 2, 'Plana Docente', 'ListadoPlanaDocenteI.php?codigo=1', 'ListadoPlanaDocenteI.php', 'PlanaDocenteI.php', 'PlanaDocenteI.php', 191, 'S', 'Contamos con docentes altamente capacitados. Profesionales con experiencia que les guiarán por la vía de la buena enseñanza.', 'Contamos con docentes altamente capacitados. Profesionales con experiencia que te guiarán por la vía de la buena enseñanza.', 0, 26, 1);

INSERT INTO sapw_opcion_web VALUES (14, 1, 'Clases Compartidas', 'Alumnos_PadresI.php', 'Alumnos_PadresI.php', 'ClasesCompartidasI.php', 'ClasesCompartidasI.php', 147, 'S', 'Docentes podrán compartir cátedras.', 'Docentes de colegios y universidades que mantienen convenio de cooperación con nuestra institución podrán compartir sus experiencias con nuestros profesores de ESPOL a fin de mejorar las técnicas de enseñanza - aprendizaje.', 0, 14, 1);

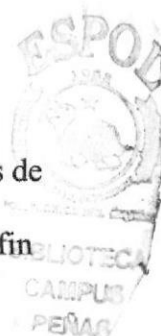
INSERT INTO sapw_opcion_web VALUES (114, 2, 'Carreras Universitarias', 'PerfilCarreraI.php?select=1', 'PerfilCarreraI.php', 'PerfilCarreraI.php', 'PerfilCarreraI.php', 511, 'V', '<< Información no Disponible >>', '
En el PROTCOM Campus Santa Elena se ofrecen las siguientes carreras modulares:

- Secretariado Ejecutivo en Sistemas de Información

- Programación de Sistemas

- Análisis de Soporte de Microcomputadores', 0, 111, 1);

INSERT INTO sapw_opcion_web VALUES (27, 2, 'Portafolio de Servicios Educativos', 'Docentes_CampusI.php', 'ConstruccionI.php', 'Docentes_CampusI.php', 'ConstruccionI.php', 358, 'S', 'Seminarios y Certificados del PROTCOM Campus Santa Elena.', 'Dirigidos al público en general, personas que no tienen conocimientos en computación y que desean ingresar a un plan de capacitación progresiva en el campo informático. También están dirigidos al personal del sector empresarial, que desea mantener una constante preparación y actualización de conocimientos, útiles para beneficio del participante y de la empresa en general.



La base de estos certificados de capacitación se originan en los seminarios de computación que dicta el PROTCOM Campus Santa Elena de manera abierta al público. Una vez que el participante aprueba un seminario, nosotros lo guiamos para que tome otros seminarios relacionados, involucrándolo en un proceso de educación continua que lo capacitará en un área específica de la informática.

 El participante está en la libertad de elegir el seminario que se adecúe a sus necesidades y posibilidades, aprobarlo y recibir su diploma respectivo. La aprobación de este seminario le acredita para la obtención futura de un "Certificado de Capacitación" de mayor jerarquía y peso académico.

La acreditación de un Certificado de Capacitación, se la obtiene cuando el estudiante haya completado un mínimo de horas de clases, que las puede obtener de un máximo de 6 seminarios con una duración de 30 horas de promedio. Un Certificado se lo puede obtener en un mínimo de 6 meses.', 0, 27, 1);

INSERT INTO sapw_opcion_web VALUES (30, 2, 'Incubadora de Ideas', 'Docentes_CampusI.php', 'Docentes_CampusI.php', 'IncubadorIdeasI.php', 'IncubadorIdeasI.php', 94, 'S', 'Colectar ideas para proyectar el desarrollo de Productos.', 'La incubación de ideas permitirá que docentes y estudiantes puedan, en conjunto, proponer temas de investigación que se cristalicen en la generación de proyectos de beneficio institucional e interinstitucional. La conformación de los grupos de trabajo, tendrá como normativa, la que el PROTCOM Campus Santa Elena estimara conveniente.', 0, 30, 1);

INSERT INTO sapw_opcion_web VALUES (110, 2, 'Directivos de la Espol', 'DirectivosI.php', 'DirectivosI.php', 'DirectivosI.php', 'DirectivosI.php', 235, 'V', '<< Información no Disponible >>', '<< Información no Disponible >>', 0, 107, 1);

INSERT INTO sapw_opcion_web VALUES (39, 33, 'Redes', 'Negocios_ComunidadI.php', 'ConstruccionI.php', 'RedesI.php', 'ConstruccionI.php', 100, 'S', 'Contamos con el mejor recurso humano, que administra, controla y organiza nuestros laboratorios, para brindar el mejor servicio a nuestros estudiantes.', 'Contamos con el mejor recurso humano, que administra, controla y organiza nuestros laboratorios, para brindar el mejor servicio a nuestros estudiantes.', 0, 39, 1);

INSERT INTO sapw_opcion_web VALUES (35, 33, 'Webmedia', 'Negocios_ComunidadI.php', 'ConstruccionI.php', 'WebMediaI.php', 'ConstruccionI.php', 91, 'S', '<< Información no disponible >>', '
Web y Multimedia de la ESPOL [WEBMEDIA] es una de las áreas que la ESPOL ha constituido para cubrir diversos requerimientos comunicacionales y de multimedios que las empresas de estos tiempos. Cada uno de nuestros productos han sido elaborados en base a estándares internacionales de desarrollo, siguiendo normas que nos han hecho acreedores a un reconcimiento sostenido en el medio empresarial, tanto privado como gubernamental, dentro y fuera del país.

Nuestros servicios se extienden a:

Presentaciones multimedias a nivel de Catálogos Digitales, Base de Datos con Multimedia, Mapas Interactivos de Planes Vitales y Logísticos, Paneos Virtuales 2D/3D e interfaces de acceso a Sistemas Informáticos.
Además desarrollo de contenidos curriculares interactivos on / off-line y otras herramientas instruccionales para el sector educativo de todo nivel.
Diseño y Desarrollo de Modelos de Negocio y Aplicaciones para el web, a nivel de redes Intranet, Extranet con conexión a Sistemas Centrales Corporativos, Tecnología HTML, DHTML, ASP/VBScript, PHP/MySQL, Java, SQL, DB2, SHOCKWAVE, Flash, Virtual Movies, Lingo / Action Scripting y demás elementos corss-plataform.
Diseño, Modelamiento y Animación 3D, CD-ROM video y

sonido digital.

WEBMEDIA conceptualiza y diseña soluciones y estrategias multimediales integrales corporativas.

INSERT INTO sapw_opcion_web VALUES (33, 3, 'Portafolio de Servicios

Profesionales', 'Negocios_ComunidadI.php', 'ConstruccionI.php',

'Negocios_ComunidadI.php', 'ConstruccionI.php', 414, 'S', 'Servicios que ofrece el PROTCOM Campus Santa Elena.', 'Le ofrecemos a Usted desde el Diseño de Sistemas de Información al Diseño de Sitios Web y Componentes Multimedia, desde Servicios de Reparación y Mantenimiento de Microcomputadores hasta el Cableado de Redes, cualquiera que sea su necesidad en PROTCOM Campus Santa Elena, hacemos la diferencia.

Ofreciendo nuestros servicios a empresas de la Península y Guayaquil, algunas de ellas son:

- COLEGIO JOHN F. KENNEDY
- UNIDAD EDUCATIVA EUGENIO ESPEJO
- COLEGIO SANTA ROSA
- BI MOT-14 MARAÑÓN
- COLEGIO GUILLERMO ORDÓÑEZ GÓMEZ
- COLEGIO PROSPERIDAD
- COLEGIO NACIONAL TÉCNICO ANCÓN
- JUNTA PARROQUIAL DE ANCÓN
- BIOAZUL
- ZAMPIVI
- CERRAMETAL
- MUNICIPIO DE SANTA ELENA
- MAR CLUB

INSERT INTO sapw_opcion_web VALUES (134, 2, 'Becas Estudiantiles', 'FondoBecaI.php', 'ConstruccionI.php', 'ConstruccionI.php', 'ConstruccionI.php', 130, 'V', 'El Fondo de Becas para el desarrollo de la Península de Santa Elena se creó en el mes de junio de 1999, como un apoyo para los estudiantes de la Península. Al este fondo pueden acceder todas las personas que cumplan ciertos requisitos que se te especificarán en esta opción.', 'El Fondo de Becas para el desarrollo de la Península de Santa Elena se creó en el mes de junio de 1999, como un apoyo para los estudiantes de la Península. Al este fondo pueden acceder todas las personas que cumplan ciertos requisitos que se te especificarán en esta opción.', 0, 21, 1);

INSERT INTO sapw_opcion_web VALUES (93, 2, 'Quiénes Somos?', 'AntecedentesI.php', 'ConstruccionI.php', 'ConstruccionI.php', 'ConstruccionI.php', 135, 'V', '<< Información no disponible >>', '<< Información no disponible >>', 0, 93, 1);

INSERT INTO sapw_opcion_web VALUES (117, 5, 'Graduaciones del PROTCOM Santa Elena', 'Ex_AlumnosIP.php', 'ConstruccionI.php', 'ConstruccionI.php', 'ConstruccionI.php', 99, 'V', '<< Información no disponible >>', '<< Información no disponible >>', 0, 205, 1);

INSERT INTO sapw_opcion_web VALUES (94, 1, 'Testimonios de Ex-Alumnos', 'TestimoniosI.php', 'ConstruccionI.php', 'ConstruccionI.php', 'ConstruccionI.php', 7, 'S', 'Podrá revisar testimonios de Ex-alumnos del PROTCOM Campus Santa Elena que han alcanzado éxito en el mundo de la Informática actualmente.', 'Podrá revisar testimonios de Ex-alumnos del PROTCOM Campus Santa Elena que han alcanzado éxito en el mundo de la Informática actualmente.', 0, 94, 1);

--Opciones del Sistema

```
INSERT INTO sapw_opcion_sistema VALUES (1, 0, 'Seguridad', 'Mnu_Seguridad', 0);
INSERT INTO sapw_opcion_sistema VALUES (2, 1, 'Auditoria del Sistema',
'Mnu_Seg_Auditoria', 1);
INSERT INTO sapw_opcion_sistema VALUES (3, 1, 'Diccionario de Búsqueda del
Portal', 'Mnu_Seg_Diccionario', 1);
INSERT INTO sapw_opcion_sistema VALUES (4, 1, 'Opciones del Portal',
'Mnu_Seg_OpcionesPortal', 1);
INSERT INTO sapw_opcion_sistema VALUES (5, 1, 'Usuario', 'Mnu_Seg_Usuario',
1);
INSERT INTO sapw_opcion_sistema VALUES (6, 1, 'Grupo de Usuarios',
'Mnu_Seg_GrupoUsuario', 1);
INSERT INTO sapw_opcion_sistema VALUES (7, 1, 'Parametros Generales',
'Mnu_Seg_Parametros', 1);
INSERT INTO sapw_opcion_sistema VALUES (8, 1, 'Información del Servidor',
'Mnu_Seg_InformacionServidor', 1);
INSERT INTO sapw_opcion_sistema VALUES (9, 1, 'Permisos por Usuarios',
'Mnu_Seg_PermisosUsuario', 1);
INSERT INTO sapw_opcion_sistema VALUES (12, 1, 'Permisos por Grupo de
Usuarios', 'Mnu_Seg_PermisosGrupo', 1);
INSERT INTO sapw_opcion_sistema VALUES (15, 0, 'Mantenimientos',
'Mnu_Mantenimientos', 0);
INSERT INTO sapw_opcion_sistema VALUES (16, 15, 'Portafolio de Servicios',
'Mnu_Man_PortafolioServicios', 1);
INSERT INTO sapw_opcion_sistema VALUES (17, 16, 'Educativos',
'Mnu_Man_Educativos', 2);
INSERT INTO sapw_opcion_sistema VALUES (18, 17, 'Planificación',
'Mnu_Man_Planificacion', 3);
INSERT INTO sapw_opcion_sistema VALUES (19, 18, 'Años', 'Mnu_Man_Anios', 4);
INSERT INTO sapw_opcion_sistema VALUES (20, 18, 'Horarios',
'Mnu_Man_Horarios', 4);
INSERT INTO sapw_opcion_sistema VALUES (21, 18, 'Materias',
'Mnu_Man_Materias', 4);
INSERT INTO sapw_opcion_sistema VALUES (22, 18, 'Laboratorios',
'Mnu_Man_Laboratorios', 4);
INSERT INTO sapw_opcion_sistema VALUES (23, 18, 'Aulas de Clase',
'Mnu_Man_AulasClase', 4);
INSERT INTO sapw_opcion_sistema VALUES (24, 18, 'Carreras Universitarias',
'Mnu_Man_CarrerasUniversitarias', 4);
INSERT INTO sapw_opcion_sistema VALUES (25, 18, 'Niveles Académicos',
'Mnu_Man_NivelesAcademicos', 4);
INSERT INTO sapw_opcion_sistema VALUES (26, 18, 'Pensum Académico',
'Mnu_Man_PensumAcademico', 4);
INSERT INTO sapw_opcion_sistema VALUES (27, 18, 'Plana Docente',
'Mnu_Man_PlanaDocente', 4);
INSERT INTO sapw_opcion_sistema VALUES (28, 18, 'Año Lectivo',
'Mnu_Man_AnioLectivo', 4);
INSERT INTO sapw_opcion_sistema VALUES (29, 18, 'Periodos de Carreras
Universitarias', 'Mnu_Man_Periodos', 4);
```



```
INSERT INTO sapw_opcion_sistema VALUES (30, 18, 'Paralelos',  
'Mnu_Man_Paralelos', 4);  
INSERT INTO sapw_opcion_sistema VALUES (31, 17, 'Registros',  
'Mnu_Man_Registros', 3);  
INSERT INTO sapw_opcion_sistema VALUES (32, 31, 'Docentes',  
'Mnu_Man_Docentes', 4);  
INSERT INTO sapw_opcion_sistema VALUES (33, 31, 'Estudiantes',  
'Mnu_Man_Estudiantes', 4);  
INSERT INTO sapw_opcion_sistema VALUES (34, 31, 'Estudiantes a Carreras  
Universitarias', 'Mnu_Man_EstudiantesCarrera', 4);  
INSERT INTO sapw_opcion_sistema VALUES (35, 31, 'Inscripción de Estudiantes',  
'Mnu_Man_InscripcionEstudiantes', 4);  
INSERT INTO sapw_opcion_sistema VALUES (36, 31, 'Notas Estudiantiles',  
'Mnu_Man_NotasEstudiantiles', 4);  
INSERT INTO sapw_opcion_sistema VALUES (37, 31, 'Deudas Estudiantiles',  
'Mnu_Man_DeudasEstudiantiles', 4);  
INSERT INTO sapw_opcion_sistema VALUES (38, 31, 'Promociones Graduadas',  
'Mnu_Man_PromocionesGraduadas', 4);  
INSERT INTO sapw_opcion_sistema VALUES (39, 31, 'Políticas del Curso',  
'Mnu_Man_PolicasCurso', 4);  
INSERT INTO sapw_opcion_sistema VALUES (40, 31, 'Capítulos de Estudio',  
'Mnu_Man_CapitulosEstudio', 4);  
INSERT INTO sapw_opcion_sistema VALUES (41, 31, 'Deberes',  
'Mnu_Man_Deberes', 4);  
INSERT INTO sapw_opcion_sistema VALUES (42, 16, 'Desarrollo de Proyectos',  
'Mnu_Man_DesarrolloProyectos', 2);  
INSERT INTO sapw_opcion_sistema VALUES (43, 42, 'Servicios',  
'Mnu_Man_Servicios', 3);  
INSERT INTO sapw_opcion_sistema VALUES (44, 42, 'Productos',  
'Mnu_Man_Productos', 3);  
INSERT INTO sapw_opcion_sistema VALUES (45, 16, 'Varios', 'Mnu_Man_Varios',  
2);  
INSERT INTO sapw_opcion_sistema VALUES (46, 45, 'Programoteca',  
'Mnu_Man_Programoteca', 3);  
INSERT INTO sapw_opcion_sistema VALUES (51, 15, 'Personal Administrativo',  
'Mnu_Man_PersonalAdministrativo', 1);  
INSERT INTO sapw_opcion_sistema VALUES (52, 51, 'Cargos', 'Mnu_Man_Cargos',  
2);  
INSERT INTO sapw_opcion_sistema VALUES (53, 51, 'Departamentos',  
'Mnu_Man_Departamentos', 2);  
INSERT INTO sapw_opcion_sistema VALUES (54, 51, 'Empresa',  
'Mnu_Man_Empresa', 2);  
INSERT INTO sapw_opcion_sistema VALUES (55, 51, 'Título Profesional',  
'Mnu_Man_TituloProfesional', 2);  
INSERT INTO sapw_opcion_sistema VALUES (56, 51, 'Persona',  
'Mnu_Man_Persona', 2);  
INSERT INTO sapw_opcion_sistema VALUES (57, 51, 'Directivos',  
'Mnu_Man_Directivos', 2);  
INSERT INTO sapw_opcion_sistema VALUES (58, 51, 'Reglamentos',  
'Mnu_Man_Reglamentos', 2);
```

```
INSERT INTO sapw_opcion_sistema VALUES (59, 51, 'Resoluciones',  
'Mnu_Man_Resoluciones', 2);  
INSERT INTO sapw_opcion_sistema VALUES (60, 51, 'Políticas de Reservación de  
Laboratorios', 'Mnu_Man_PolíticasReservacion', 2);  
INSERT INTO sapw_opcion_sistema VALUES (61, 15, 'Servicios al Público',  
'Mnu_Man_ServiciosPublico', 1);  
INSERT INTO sapw_opcion_sistema VALUES (62, 61, 'Certificados',  
'Mnu_Man_Certificados', 2);  
INSERT INTO sapw_opcion_sistema VALUES (63, 62, 'Planificación',  
'Mnu_Man_PlanificacionCertificados', 3);  
INSERT INTO sapw_opcion_sistema VALUES (64, 62, 'Registros',  
'Mnu_Man_RegistrosCertificados', 3);  
INSERT INTO sapw_opcion_sistema VALUES (65, 62, 'Observaciones',  
'Mnu_Man_Observaciones', 3);  
INSERT INTO sapw_opcion_sistema VALUES (68, 61, 'Catálogo Digital',  
'Mnu_Man_CatalogoDitigal', 2);  
INSERT INTO sapw_opcion_sistema VALUES (69, 61, 'Clientes',  
'Mnu_Man_Clientes', 2);  
INSERT INTO sapw_opcion_sistema VALUES (70, 61, 'Eventos',  
'Mnu_Man_Eventos', 2);  
INSERT INTO sapw_opcion_sistema VALUES (71, 15, 'Servicios del Sitio',  
'Mnu_Man_ServiciosSitio', 1);  
INSERT INTO sapw_opcion_sistema VALUES (72, 71, 'Anuncios',  
'Mnu_Man_Anuncios', 2);  
INSERT INTO sapw_opcion_sistema VALUES (73, 71, 'Curriculum Vitae',  
'Mnu_Man_CurriculumVitae', 2);  
INSERT INTO sapw_opcion_sistema VALUES (74, 71, 'Foros de Discusión',  
'Mnu_Man_ForosDiscusion', 2);  
INSERT INTO sapw_opcion_sistema VALUES (75, 71, 'Clasificaciones de Foros de  
Discusión', 'Mnu_Man_ClasificacionesForos', 2);  
INSERT INTO sapw_opcion_sistema VALUES (76, 71, 'Preguntas al Foro de  
Discusión', 'Mnu_Man_PreguntasForo', 2);  
INSERT INTO sapw_opcion_sistema VALUES (77, 71, 'Responder Preguntas del  
Forum', 'Mnu_Man_ResponderPreguntaForum', 2);  
INSERT INTO sapw_opcion_sistema VALUES (78, 71, 'Noticias T.I',  
'Mnu_Man_NoticiasTI', 2);  
INSERT INTO sapw_opcion_sistema VALUES (79, 71, 'Recurso Humano disponible',  
'Mnu_Man_RecursoHumano', 2);  
INSERT INTO sapw_opcion_sistema VALUES (80, 71, 'Solicitud de Personal',  
'Mnu_Man_SolicitudPersonal', 2);  
INSERT INTO sapw_opcion_sistema VALUES (81, 71, 'Tips Academicos',  
'Mnu_Man_TipsAcademicos', 2);  
INSERT INTO sapw_opcion_sistema VALUES (82, 0, 'Procesos', 'Mnu_Procesos', 0);  
INSERT INTO sapw_opcion_sistema VALUES (83, 82, 'Activar Período Actual por  
Carrera Universitaria', 'Mnu_Pro_ActivarPeriodo', 1);  
INSERT INTO sapw_opcion_sistema VALUES (84, 82, 'Activar Año Actual',  
'Mnu_Pro_ActivarAnio', 1);  
INSERT INTO sapw_opcion_sistema VALUES (85, 82, 'Activar Buzón',  
'Mnu_Pro_ActivarBuzon', 1);
```



```
INSERT INTO sapw_opcion_sistema VALUES (86, 82, 'Activar Preguntas del Foro de
Discusión', 'Mnu_Pro_ActivarPreguntas', 1);
INSERT INTO sapw_opcion_sistema VALUES (87, 82, 'Activar Respuestas del Foro
de Discusión', 'Mnu_Pro_ActivarRespuestas', 1);
INSERT INTO sapw_opcion_sistema VALUES (88, 82, 'Revisión de Solicitud de
Seminarios', 'Mnu_Pro_RevisionCapacitacion', 1);
INSERT INTO sapw_opcion_sistema VALUES (89, 82, 'Revisión de Solicitudes de
Reservación de Laboratorios', 'Mnu_Pro_RevisarSolicitudReservacion', 1);
INSERT INTO sapw_opcion_sistema VALUES (90, 82, 'Revisión de Solicitudes de
Pasantías', 'Mnu_Pro_RevisarSolicitudPasantia', 1);
INSERT INTO sapw_opcion_sistema VALUES (91, 82, 'Revisión del Buzón de
Sugerencias', 'Mnu_Pro_RevisionBuzon', 1);
INSERT INTO sapw_opcion_sistema VALUES (92, 82, 'Cierre Periodo Lectivo',
'Mnu_Pro_CierrePeriodo', 1);
INSERT INTO sapw_opcion_sistema VALUES (93, 82, 'Eliminación de Archivos
Temporales', 'Mnu_Pro_EliminacionArchivos', 1);
INSERT INTO sapw_opcion_sistema VALUES (94, 82, 'Ejecutar Archivo Estudiantes',
'Mnu_Pro_EjecutarArchivoEstudiantes', 1);
INSERT INTO sapw_opcion_sistema VALUES (95, 82, 'Ejercutar Archivo Docentes',
'Mnu_Pro_EjecutarArchivoDocentes', 1);
INSERT INTO sapw_opcion_sistema VALUES (96, 82, 'Estudiantes Egresados',
'Mnu_Pro_EstudiantesEgresados', 1);
INSERT INTO sapw_opcion_sistema VALUES (97, 82, 'Inscripción a Foros de
Discusión', 'Mnu_Pro_InscripcionForos', 1);
INSERT INTO sapw_opcion_sistema VALUES (98, 0, 'Consultas y Reportes',
'Mnu_Consultas', 0);
INSERT INTO sapw_opcion_sistema VALUES (99, 98, 'Deudas Estudiantiles',
'Mnu_Con_DeudasEstudiantiles', 1);
INSERT INTO sapw_opcion_sistema VALUES (100, 98, 'Departamentos',
'Mnu_Con_Departamentos', 1);
INSERT INTO sapw_opcion_sistema VALUES (101, 98, 'Pensum Académico',
'Mnu_Con_PensumAcademico', 1);
INSERT INTO sapw_opcion_sistema VALUES (102, 98, 'Historial Estudiantil',
'Mnu_Con_HistorialEstudiantil', 1);
INSERT INTO sapw_opcion_sistema VALUES (103, 98, 'Estudiantes',
'Mnu_Con_Estudiantes', 1);
INSERT INTO sapw_opcion_sistema VALUES (104, 98, 'Plana Docente',
'Mnu_Con_PlanaDocente', 1);
INSERT INTO sapw_opcion_sistema VALUES (106, 98, 'Certificados',
'Mnu_Con_Certificados', 1);
INSERT INTO sapw_opcion_sistema VALUES (107, 98, 'Promociones Graduadas',
'Mnu_Con_PromocionesGraduadas', 1);
INSERT INTO sapw_opcion_sistema VALUES (108, 98, 'Pasantías',
'Mnu_Con_Pasantias', 1);
INSERT INTO sapw_opcion_sistema VALUES (109, 108, 'Solicitudes',
'Mnu_Con_Solicitudes', 2);
INSERT INTO sapw_opcion_sistema VALUES (110, 108, 'Aprobadas',
'Mnu_Con_Aprobadas', 2);
INSERT INTO sapw_opcion_sistema VALUES (111, 98, 'Opciones mas visitadas',
'Mnu_Con_Opciones', 1);
```

```
INSERT INTO sapw_opcion_sistema VALUES (112, 98, 'SubOpciones mas visitadas',
'Mnu_Con_SubOpciones', 1);
INSERT INTO sapw_opcion_sistema VALUES (11, 9, 'Portal Protcom Santa Elena',
'Mnu_Seg_PermUsuaPortal', 2);
INSERT INTO sapw_opcion_sistema VALUES (14, 12, 'Portal Protcom Santa Elena',
'Mnu_Seg_PermGrupoPortal', 2);
INSERT INTO sapw_opcion_sistema VALUES (10, 9, 'Sistema Portal Protcom Santa
Elena', 'Mnu_Seg_PermUsuaSistema', 2);
INSERT INTO sapw_opcion_sistema VALUES (13, 12, 'Sistema Portal Protcom Santa
Elena', 'Mnu_Seg_PermGrupoSistema', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 1, 0, 'Seguridad',
'Mnu_Seguridad', 0);
INSERT INTO SAPW_Opcion_Sistema VALUES( 2, 1, 'Auditoria del Sistema',
'Mnu_Seg_Auditoria', 1);
INSERT INTO SAPW_Opcion_Sistema VALUES( 3, 1, 'Diccionario de Búsqueda
del Portal', 'Mnu_Seg_Diccionario', 1);
INSERT INTO SAPW_Opcion_Sistema VALUES( 4, 1, 'Opciones del Portal',
'Mnu_Seg_OpcionesPortal', 1);
INSERT INTO SAPW_Opcion_Sistema VALUES( 5, 1, 'Usuario',
'Mnu_Seg_Usuario', 1);
INSERT INTO SAPW_Opcion_Sistema VALUES( 6, 1, 'Grupo de Usuarios',
'Mnu_Seg_GrupoUsuario', 1);
INSERT INTO SAPW_Opcion_Sistema VALUES( 7, 1, 'Parametros Generales',
'Mnu_Seg_Parametros', 1);
INSERT INTO SAPW_Opcion_Sistema VALUES( 8, 1, 'Información del Servidor',
'Mnu_Seg_InformacionServidor', 1);
INSERT INTO SAPW_Opcion_Sistema VALUES( 9, 1, 'Permisos por Usuarios',
'Mnu_Seg_PermisosUsuario', 1);
INSERT INTO SAPW_Opcion_Sistema VALUES( 10, 9, 'Sistema Portal Protcom',
'Mnu_Seg_PermUsuaSistema', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 11, 9, 'Portal Protcom',
'Mnu_Seg_PermUsuaPortal', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 12, 1, 'Permisos por Grupo de
Usuarios', 'Mnu_Seg_PermisosGrupo', 1);
INSERT INTO SAPW_Opcion_Sistema VALUES( 13, 12, 'Sistema Portal
Protcom', 'Mnu_Seg_PermGrupoSistema', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 14, 12, 'Portal Protcom',
'Mnu_Seg_PermGrupoPortal', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 15, 0, 'Mantenimientos',
'Mnu_Mantenimientos', 0);
INSERT INTO SAPW_Opcion_Sistema VALUES( 16, 15, 'Portafolio de Servicios',
'Mnu_Man_PortafolioServicios', 1);
--' Root > Mantenimientos > Portafolio de Servicios...
INSERT INTO SAPW_Opcion_Sistema VALUES( 17, 16, 'Educativos',
'Mnu_Man_Educativos', 2);
--' Root > Mantenimientos > Portafolio de Servicios > Educativos...
INSERT INTO SAPW_Opcion_Sistema VALUES( 18, 17, 'Planificación',
'Mnu_Man_Planificacion', 3);
INSERT INTO SAPW_Opcion_Sistema VALUES( 19, 18, 'Años', 'Mnu_Man_Anios',
4);
```



```
INSERT INTO SAPW_Opcion_Sistema VALUES( 20, 18, 'Horarios',  
'Mnu_Man_Horarios', 4);  
INSERT INTO SAPW_Opcion_Sistema VALUES( 21, 18, 'Materias',  
'Mnu_Man_Materias', 4);  
INSERT INTO SAPW_Opcion_Sistema VALUES( 22, 18, 'Laboratorios',  
'Mnu_Man_Laboratorios', 4);  
INSERT INTO SAPW_Opcion_Sistema VALUES( 23, 18, 'Aulas de Clase',  
'Mnu_Man_AulasClase', 4);  
INSERT INTO SAPW_Opcion_Sistema VALUES( 24, 18, 'Carreras Universitarias',  
'Mnu_Man_CarrerasUniversitarias', 4);  
INSERT INTO SAPW_Opcion_Sistema VALUES( 25, 18, 'Niveles Académicos',  
'Mnu_Man_NivelesAcademicos', 4);  
INSERT INTO SAPW_Opcion_Sistema VALUES( 26, 18, 'Pensum Académico',  
'Mnu_Man_PensumAcademico', 4);  
INSERT INTO SAPW_Opcion_Sistema VALUES( 27, 18, 'Plana Docente',  
'Mnu_Man_PlanaDocente', 4);  
INSERT INTO SAPW_Opcion_Sistema VALUES( 28, 18, 'Año Lectivo',  
'Mnu_Man_AnioLectivo', 4);  
INSERT INTO SAPW_Opcion_Sistema VALUES( 29, 18, 'Periodos de Carreras  
Universitarias', 'Mnu_Man_Periodos', 4);  
INSERT INTO SAPW_Opcion_Sistema VALUES( 30, 18, 'Paralelos',  
'Mnu_Man_Paralelos', 4);  
INSERT INTO SAPW_Opcion_Sistema VALUES( 31, 18, 'Distribución de Horarios  
por Paralelos', 'Mnu_Man_DistribucionHorarios', 4);  
INSERT INTO SAPW_Opcion_Sistema VALUES( 32, 17, 'Registros',  
'Mnu_Man_Registros', 3);  
INSERT INTO SAPW_Opcion_Sistema VALUES( 33, 32, 'Docentes',  
'Mnu_Man_Docentes', 4);  
INSERT INTO SAPW_Opcion_Sistema VALUES( 34, 32, 'Estudiantes',  
'Mnu_Man_Estudiantes', 4);  
INSERT INTO SAPW_Opcion_Sistema VALUES( 35, 32, 'Estudiantes a Carreras  
Universitarias', 'Mnu_Man_EstudiantesCarrera', 4);  
INSERT INTO SAPW_Opcion_Sistema VALUES( 36, 32, 'Inscripción de  
Estudiantes', 'Mnu_Man_InscripcionEstudiantes', 4);  
INSERT INTO SAPW_Opcion_Sistema VALUES( 37, 32, 'Notas Estudiantiles',  
'Mnu_Man_NotasEstudiantiles', 4);  
INSERT INTO SAPW_Opcion_Sistema VALUES( 38, 32, 'Deudas Estudiantiles',  
'Mnu_Man_DeudasEstudiantiles', 4);  
INSERT INTO SAPW_Opcion_Sistema VALUES( 39, 32, 'Promociones Graduadas',  
'Mnu_Man_PromocionesGraduadas', 4);  
INSERT INTO SAPW_Opcion_Sistema VALUES( 40, 32, 'Forum por Paralelos',  
'Mnu_Man_ForumParalelo', 4);  
INSERT INTO SAPW_Opcion_Sistema VALUES( 41, 32, 'Preguntas para el Forum',  
'Mnu_Man_PreguntasForumParalelo', 4);  
INSERT INTO SAPW_Opcion_Sistema VALUES( 42, 32, 'Políticas del Curso',  
'Mnu_Man_PoliticasCurso', 4);  
INSERT INTO SAPW_Opcion_Sistema VALUES( 43, 32, 'Capítulos de Estudio',  
'Mnu_Man_CapitulosEstudio', 4);  
INSERT INTO SAPW_Opcion_Sistema VALUES( 44, 32, 'Deberes',  
'Mnu_Man_Deberes', 4);
```



```
INSERT INTO SAPW_Opcion_Sistema VALUES( 45, 16, 'Desarrollo de Proyectos',
'Mnu_Man_DesarrolloProyectos', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 46, 45, 'Servicios',
'Mnu_Man_Servicios', 3);
INSERT INTO SAPW_Opcion_Sistema VALUES( 47, 45, 'Productos',
'Mnu_Man_Productos', 3);
INSERT INTO SAPW_Opcion_Sistema VALUES( 48, 16, 'Varios',
'Mnu_Man_Varios', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 49, 48, 'Programoteca',
'Mnu_Man_Programoteca', 3);
INSERT INTO SAPW_Opcion_Sistema VALUES( 54, 15, 'Personal Administrativo',
'Mnu_Man_PersonalAdministrativo', 1);
--' Root > Mantenimientos > Personal Administrativo...
INSERT INTO SAPW_Opcion_Sistema VALUES( 55, 54, 'Cargos',
'Mnu_Man_Cargos', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 56, 54, 'Departamentos',
'Mnu_Man_Departamentos', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 57, 54, 'Empresa',
'Mnu_Man_Empresa', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 58, 54, 'Titulo Profesional',
'Mnu_Man_TituloProfesional', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 59, 54, 'Persona',
'Mnu_Man_Persona', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 60, 54, 'Directivos',
'Mnu_Man_Directivos', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 61, 54, 'Reglamentos',
'Mnu_Man_Reglamentos', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 62, 54, 'Resoluciones',
'Mnu_Man_Resoluciones', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 63, 54, 'Políticas de Reservación
de Laboratorios', 'Mnu_Man_PolíticasReservacion', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 64, 15, 'Servicios al Público',
'Mnu_Man_ServiciosPublico', 1);
INSERT INTO SAPW_Opcion_Sistema VALUES( 65, 64, 'Certificados',
'Mnu_Man_Certificados', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 66, 65, 'Planificación',
'Mnu_Man_PlanificacionCertificados', 3);
INSERT INTO SAPW_Opcion_Sistema VALUES( 67, 65, 'Registros',
'Mnu_Man_RegistrosCertificados', 3);
INSERT INTO SAPW_Opcion_Sistema VALUES( 68, 65, 'Observaciones',
'Mnu_Man_Observaciones', 3);
INSERT INTO SAPW_Opcion_Sistema VALUES( 70, 64, 'Documentos Solicitados
para Certificados', 'Mnu_Man_Documentos', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 71, 64, 'Catalogo Digital',
'Mnu_Man_CatalogoDitigal', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 72, 64, 'Clientes',
'Mnu_Man_Clientes', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 73, 64, 'Eventos',
'Mnu_Man_Eventos', 2);
```

```
INSERT INTO SAPW_Opcion_Sistema VALUES( 74, 15, 'Servicios del Sitio',
'Mnu_Man_ServiciosSitio', 1);
INSERT INTO SAPW_Opcion_Sistema VALUES( 75, 74, 'Anuncios',
'Mnu_Man_Anuncios', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 76, 74, 'Curriculum Vitae',
'Mnu_Man_CurriculumVitae', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 77, 74, 'Foros de Discusión',
'Mnu_Man_ForosDiscusion', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 78, 74, 'Inscripcion a Foros de
Discusión', 'Mnu_Man_InscripcionForos', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 79, 74, 'Responder Preguntas del
Forum', 'Mnu_Man_ResponderPreguntaForum', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 80, 74, 'Noticias T.I',
'Mnu_Man_NoticiasTI', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 81, 74, 'Recurso Humano
disponible', 'Mnu_Man_RecursoHumano', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 82, 74, 'Solicitud de Personal',
'Mnu_Man_SolicitudPersonal', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 83, 74, 'Tips Academicos',
'Mnu_Man_TipsAcademicos', 2);
INSERT INTO SAPW_Opcion_Sistema VALUES( 84, 0, 'Procesos', 'Mnu_Procesos',
0);
INSERT INTO SAPW_Opcion_Sistema VALUES( 85, 84, 'Activar Periodo Actual por
Carrera Universitaria', 'Mnu_Pro_ActivarPeriodo', 1);
INSERT INTO SAPW_Opcion_Sistema VALUES( 86, 84, 'Activar Año Actual',
'Mnu_Pro_ActivarAnio', 1);
INSERT INTO SAPW_Opcion_Sistema VALUES( 87, 84, 'Revisión de Buzón',
'Mnu_Pro_RevisionBuzon', 1);
INSERT INTO SAPW_Opcion_Sistema VALUES( 88, 84, 'Activar Buzón',
'Mnu_Pro_ActivarBuzon', 1);
INSERT INTO SAPW_Opcion_Sistema VALUES( 89, 84, 'Revisar Solicitud
Reservación Laboratorios', 'Mnu_Pro_RevisarSolicitudReservacion', 1);
INSERT INTO SAPW_Opcion_Sistema VALUES( 90, 84, 'Activar Preguntas Forum',
'Mnu_Pro_ActivarPreguntas', 1);
INSERT INTO SAPW_Opcion_Sistema VALUES( 91, 84, 'Activar Respuestas
Forum', 'Mnu_Pro_ActivarRespuestas', 1);
INSERT INTO SAPW_Opcion_Sistema VALUES( 92, 84, 'Revisar Solicitud Pasantia',
'Mnu_Pro_RevisarSolicitudPasantia', 1);
INSERT INTO SAPW_Opcion_Sistema VALUES( 93, 84, 'Cierre Periodo Lectivo',
'Mnu_Pro_CierrePeriodo', 1);
INSERT INTO SAPW_Opcion_Sistema VALUES( 94, 84, 'Eliminación de Archivos
Temporales', 'Mnu_Pro_EliminacionArchivos', 1);
INSERT INTO SAPW_Opcion_Sistema VALUES( 95, 84, 'Ejecutar Archivo
Estudiantes', 'Mnu_Pro_EjecutarArchivoEstudiantes', 1);
INSERT INTO SAPW_Opcion_Sistema VALUES( 96, 84, 'Ejercutar Archivo
Docentes', 'Mnu_Pro_EjecutarArchivoDocentes', 1);
INSERT INTO SAPW_Opcion_Sistema VALUES( 97, 84, 'Estudiantes Egresados',
'Mnu_Pro_EstudiantesEgresados', 1);
INSERT INTO SAPW_Opcion_Sistema VALUES( 98, 0, 'Consultas y Reportes',
'Mnu_Consultas', 0);
```

```
INSERT INTO SAPW_Opcion_Sistema VALUES( 99, 98, 'Deudas Estudiantiles',  
'Mnu_Con_DeudasEstudiantiles', 1);  
INSERT INTO SAPW_Opcion_Sistema VALUES(100, 98, 'Departamentos',  
'Mnu_Con_Departamentos', 1);  
INSERT INTO SAPW_Opcion_Sistema VALUES(101, 98, 'División',  
'Mnu_Con_Division', 1);  
INSERT INTO SAPW_Opcion_Sistema VALUES(102, 98, 'Pensum Académico',  
'Mnu_Con_PensumAcademico', 1);  
INSERT INTO SAPW_Opcion_Sistema VALUES(103, 98, 'Historial Estudiantil',  
'Mnu_Con_HistorialEstudiantil', 1);  
INSERT INTO SAPW_Opcion_Sistema VALUES(104, 98, 'Estudiantes',  
'Mnu_Con_Estudiantes', 1);  
INSERT INTO SAPW_Opcion_Sistema VALUES(105, 98, 'Plana Docente',  
'Mnu_Con_PlanaDocente', 1);  
INSERT INTO SAPW_Opcion_Sistema VALUES(107, 98, 'Certificados',  
'Mnu_Con_Certificados', 1);  
INSERT INTO SAPW_Opcion_Sistema VALUES(108, 98, 'Promociones Graduadas',  
'Mnu_Con_PromocionesGraduadas', 1);  
INSERT INTO SAPW_Opcion_Sistema VALUES(109, 98, 'Pasantias',  
'Mnu_Con_Pasantias', 1);  
INSERT INTO SAPW_Opcion_Sistema VALUES(110, 109, 'Solicitudes',  
'Mnu_Con_Solicitudes', 2);  
INSERT INTO SAPW_Opcion_Sistema VALUES(111, 109, 'Aprobadas',  
'Mnu_Con_Aprobadas', 2);
```

--Grupos de Usuarios

```
INSERT INTO SAPW_Grupo_Usuario VALUES (0, 1, 'Coordinador General', 1, 0);  
INSERT INTO SAPW_Grupo_Usuario VALUES  
(NEXTVAL('SAPW_S_Grupo_Usuario'), 1, 'Alumnos & Padres', 2, 0);  
INSERT INTO SAPW_Grupo_Usuario VALUES  
(NEXTVAL('SAPW_S_Grupo_Usuario'), 1, 'Docentes & Campus', 2, 0);  
INSERT INTO SAPW_Grupo_Usuario VALUES  
(NEXTVAL('SAPW_S_Grupo_Usuario'), 1, 'Negocios & Comunidad', 2, 0);  
INSERT INTO SAPW_Grupo_Usuario VALUES  
(NEXTVAL('SAPW_S_Grupo_Usuario'), 1, 'E - Education', 2, 0);  
INSERT INTO SAPW_Grupo_Usuario VALUES  
(NEXTVAL('SAPW_S_Grupo_Usuario'), 1, 'WebMaster', 1, 0);  
INSERT INTO SAPW_Grupo_Usuario VALUES  
(NEXTVAL('SAPW_S_Grupo_Usuario'), 1, 'Alumnos & Padres Externos', 2, 0);  
INSERT INTO SAPW_Grupo_Usuario VALUES  
(NEXTVAL('SAPW_S_Grupo_Usuario'), 1, 'Docentes & Campus Externos', 2, 0);  
INSERT INTO SAPW_Grupo_Usuario VALUES  
(NEXTVAL('SAPW_S_Grupo_Usuario'), 1, 'Negocios & Comunidad Externos', 2, 0);
```

--Títulos Profesionales

```
INSERT INTO sapw_titulo_profesional VALUES (1, 'Sin Título Profesional',  
'Ninguna', '');  
INSERT INTO sapw_titulo_profesional VALUES (2, 'Analista de Sistemas', 'Ninguna',  
'Anl.');
```

```

INSERT INTO sapw_titulo_profesional VALUES (4, 'Tecnólogo en Diseño Grafico',
'Ninguna', 'Tcnlg. ');
INSERT INTO sapw_titulo_profesional VALUES (5, 'Economista', 'Ninguna', 'Econ. ');
INSERT INTO sapw_titulo_profesional VALUES (6, 'Ingeniero Comercial', 'Ninguna',
'Ing. ');
INSERT INTO sapw_titulo_profesional VALUES (7, 'Ingeniero Industrial', 'Ninguna',
'Ing. ');
INSERT INTO sapw_titulo_profesional VALUES (8, 'Ingeniero Electronico',
'Ninguna', 'Ing. ');
INSERT INTO sapw_titulo_profesional VALUES (9, 'Ingeniero en Sistemas',
'Ninguna', 'Ing. ');
INSERT INTO sapw_titulo_profesional VALUES (10, 'Licenciada en Ciencias de la
Educación', 'Ninguna', 'Lcda. ');
INSERT INTO sapw_titulo_profesional VALUES (11, 'Tecnólogo en Electricidad',
'Ninguna', 'Tcnlg. ');
INSERT INTO sapw_titulo_profesional VALUES (12, 'Licenciada en Educación',
'Ninguna', 'Lcda. ');
INSERT INTO sapw_titulo_profesional VALUES (13, 'Ingeniero en Minas y
Petróleos', 'Ninguna', 'Ing. ');
INSERT INTO sapw_titulo_profesional VALUES (14, 'Ingeniero en Computación',
'Ninguna', 'Ing. ');
INSERT INTO sapw_titulo_profesional VALUES (15, 'Contador Público Asociado',
'Ninguna', 'CPA. ');
INSERT INTO sapw_titulo_profesional VALUES (16, 'Licenciado en Artes Visuales',
'Ninguna', 'Lcdo. ');
INSERT INTO sapw_titulo_profesional VALUES (18, 'Licenciado en Historia',
'Ninguna', 'Lcdo. ');
INSERT INTO sapw_titulo_profesional VALUES (19, 'Ingeniero Civil', 'Ninguna',
'Ing. ');
INSERT INTO sapw_titulo_profesional VALUES (20, 'Psicologa Clinica', 'Ninguna',
'Psico. ');
INSERT INTO sapw_titulo_profesional VALUES (21, 'Licenciado en Bellas Artes',
'Ninguna', 'Lcdo. ');
INSERT INTO sapw_titulo_profesional VALUES (22, 'Arquitecto', 'Ninguna', 'Arqt. ');
INSERT INTO sapw_titulo_profesional VALUES (23, 'Máster of Arts en Filosofía',
'Ninguna', 'Msc. ');
INSERT INTO sapw_titulo_profesional VALUES (24, 'PHD', 'Doctorado', 'PHD');
INSERT INTO sapw_titulo_profesional VALUES (25, 'Programador de Sistemas',
'Ninguna', 'PROG. ');
INSERT INTO sapw_titulo_profesional VALUES (3, 'Licenciada en Sistemas de
Información', 'Ninguna', 'LSI. ');
INSERT INTO sapw_titulo_profesional VALUES (17, 'Licenciada en Asignaturas
Secretariales', 'Ninguna', 'Lcda. ');

```

4.1.9 Creación de Usuarios Predeterminados.

--Usuario WebMaster

```

INSERT INTO sapw_usuario VALUES ('CoordinadorGeneral', 1, '0910506450', 5,
'evydelpozo', NULL, 1, 0, 0, 0)

```


--Permisos al Usuario Coordinador General.

```
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 1, 0, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 2, 1, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 3, 1, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 4, 1, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 5, 1, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 6, 1, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 7, 1, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 8, 1, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 9, 1, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 10, 9, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 11, 9, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 12, 1, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 13, 12, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 14, 12, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 15, 0, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 16, 15, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 17, 16, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 18, 17, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 19, 18, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 20, 18, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 21, 18, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 22, 18, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 23, 18, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 24, 18, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 25, 18, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 26, 18, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 27, 18, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 28, 18, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 29, 18, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 30, 18, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 31, 17, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 32, 31, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 33, 31, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 34, 31, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 35, 31, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 36, 31, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 37, 31, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 38, 31, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 39, 31, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 40, 31, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 41, 31, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 42, 16, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 43, 42, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 44, 42, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 45, 16, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 46, 45, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 51, 15, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 52, 51, 1);
```


[illegible]

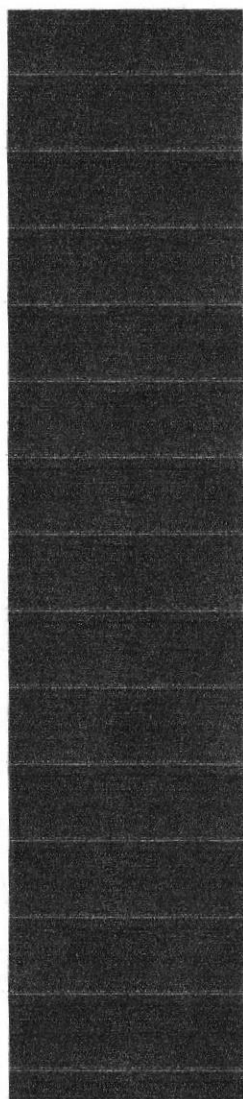
```
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 106, 98, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 107, 98, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 108, 98, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 109, 108, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 110, 108, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 111, 98, 1);
INSERT INTO sapw_perfil_usuario VALUES ('CoordinadorGeneral', 112, 98, 1);
```

--Permisos al Usuario WebMaster.

```
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 1, 0, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 2, 1, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 3, 1, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 4, 1, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 5, 1, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 6, 1, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 7, 1, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 8, 1, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 9, 1, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 10, 9, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 11, 9, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 12, 1, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 13, 12, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 14, 12, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 15, 0, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 16, 15, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 17, 16, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 18, 17, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 19, 18, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 20, 18, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 21, 18, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 22, 18, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 23, 18, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 24, 18, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 25, 18, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 26, 18, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 27, 18, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 28, 18, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 29, 18, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 30, 18, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 31, 17, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 32, 31, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 33, 31, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 34, 31, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 35, 31, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 36, 31, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 37, 31, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 38, 31, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 39, 31, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 40, 31, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 41, 31, 1);
```

```
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 42, 16, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 43, 42, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 44, 42, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 45, 16, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 46, 45, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 51, 15, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 52, 51, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 53, 51, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 54, 51, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 55, 51, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 56, 51, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 57, 51, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 58, 51, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 59, 51, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 60, 51, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 61, 15, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 62, 61, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 63, 62, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 64, 62, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 65, 62, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 68, 61, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 69, 61, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 70, 61, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 71, 15, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 72, 71, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 73, 71, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 74, 71, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 75, 71, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 76, 71, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 77, 71, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 78, 71, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 79, 71, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 80, 71, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 81, 71, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 82, 0, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 83, 82, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 84, 82, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 85, 82, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 86, 82, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 87, 82, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 88, 82, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 89, 82, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 90, 82, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 91, 82, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 92, 82, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 93, 82, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 94, 82, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 95, 82, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 96, 82, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 97, 82, 1);
```

```
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 98, 0, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 99, 98, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 100, 98, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 101, 98, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 102, 98, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 103, 98, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 104, 98, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 106, 98, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 107, 98, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 108, 98, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 109, 108, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 110, 108, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 111, 98, 1);
INSERT INTO sapw_perfil_usuario VALUES ('WebMaster', 112, 98, 1);
```

CAPÍTULO 5.

Distribución de las Clases correspondientes al Nivel de las Capas.



En este capítulo, se muestra como se encuentra organizada cada una de las clases por cada capa en el Portal Web.

5 DISTRIBUCIÓN DE LAS CLASES CORRESPONDIENTES A SU NIVEL DE CAPA

5.1 DESCRIPCIÓN GENERAL

La aplicación web está implementada con tres clases: **de datos**, **de negocios** y **de usuario**, en donde se aplica la lógica de acceso a datos, regla del negocio o políticas del negocio y de la interfaz al usuario, respectivamente. Los primeros encapsulan el acceso a los datos, donde se incluyen o realizan peticiones de inserción, modificación, eliminación y consulta dirigida a la base de datos.

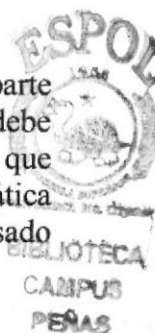
Los componentes de negocios encapsulan las reglas de negocios que son un conjunto de políticas, que definen como la empresa hace su negocio, es decir, restricciones y validaciones que deberían verse reflejada en la base de datos.

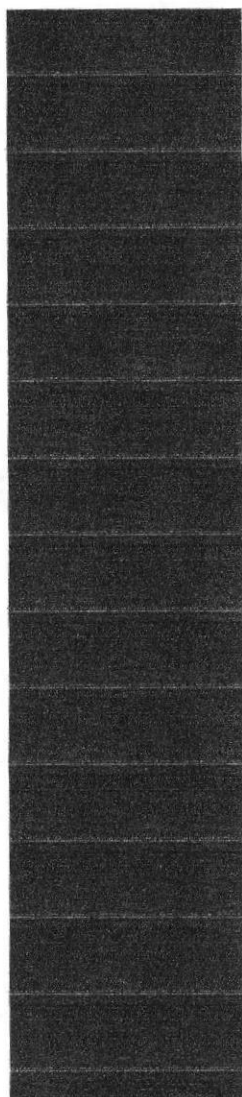
La interfaz de usuario final como lo sabemos que es el navegador pero sea creado una clase el cual separa procesos interactivos que pertenecen a la capa final de usuario evitando esta forma crear métodos redundantes.

Todos los componentes de la aplicación son de tipo **en proceso** e implementados como archivos con extensión .JOC, cuyo instanciamiento está a cargo de una capa llamada **WebSiteProtcom0000.Joc**, la cual contiene definición de variables e instancia mientras que las clases correspondientemente a cada una de las capas siendo este un proceso contenedor orientado a componentes, a continuación se detalla el resto de archivos con su correspondiente capa:

- WebSiteProtcom0001.Joc** clase de base de datos.
- WebSiteProtcom0002.Joc** clase de reglas de negocio.
- WebSiteProtcom0003.Joc** clase de interfaz de usuario.
- WebSiteProtcom0004.Joc** nivel de métodos no implementados.

Todo esto se encuentra acompañado de un proceso servidor web, siendo éste la parte fundamental de la arquitectura del sitio. El usuario que desea usar el sistema debe levantar el Web Browser y visitar la URL. El Web Browser es un proceso cliente que envía un requerimiento <http://> al servidor, la solicitud es enviada por una página estática HTML o P. H. P. que posteriormente será usado por el navegador y para ello será usado apache web server.





CAPÍTULO 6.

Instalación, y configuración de las aplicaciones.



En este capítulo, se muestra como se debe instalar y configurar cada una de las aplicaciones correspondientes para el Portal Web.

6 INSTALACIÓN Y CONFIGURACIÓN DE LAS APLICACIONES

6.1 DESCRIPCIÓN GENERAL

El portal web está realizado bajo plataformas independientes, para ello se necesitan las siguientes aplicaciones:

- Psqldb-7.x.tar
- Libiodbc-3.x.tar
- Httpd-2.x.tar
- Php-4.x.tar
- PostgreSQL-7.x.tar

Se recomienda trabajar conversiones estables.

Psqldb-7.x.tar

Para instalar esta aplicación sólo necesita ejecutar **“./configure”** pues no necesita ningún parámetro adicional.

Libiodbc-3.x.tar

Para instalar las librerías IODBC, usted debe ejecutar **“./configure”** y añadir los parámetros que indiquen donde esté desea instalar las librerías.

Httpd-2.x.tar

Para compilar el apache usted necesita realizar la siguiente:

- ✓ cd httpd_x.x.x
- ✓ ./configure --prefix=/www --enable-so
- ✓ make
- ✓ make install

Esta forma especificamos del directorio donde se va instalar el Web Server y se dejan habilitadas las librerías independientes.

Php-4.x.tar

Para construir el PHP se deben indicar los siguientes parámetros:

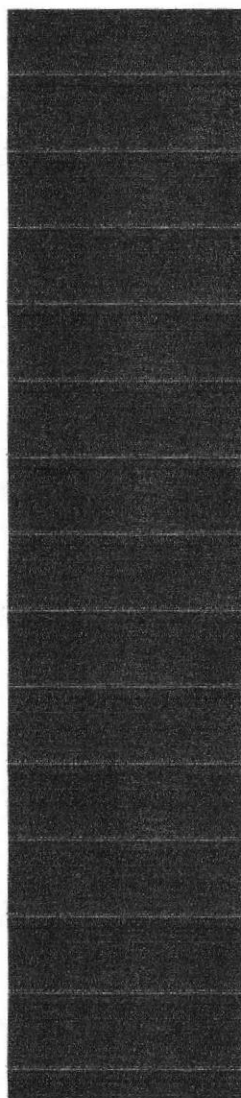
- a. cd ../php-x.x.x
- b. ./configure --with-iodbc=/usr/local/src/odbcsrc \
- with-apxs2=/www/bin/apxs

- c. make
- d. make install

Cada una de las aplicaciones pueden ser configuradas de una forma especial, dependiendo de las necesidades del usuario, si usted dese obtener el listado de los parámetros de configuración en Linux por cada una de las aplicaciones, usted debe ejecutar la siguiente línea de comando:

```
./configure --help > archivo.txt
```

De esta se obtendrá una lista de las posibles compilaciones que pueden ser utilizadas por el usuario y/o programador.



CAPÍTULO 7.

Instalación y configuración del Servidor para la conexión LDAP.



A continuación se detallan aspectos referentes a la instalación de los paquetes necesarios para la utilización del servicio LDAP en el servidor Linux y su configuración.

7 INSTALACIÓN Y CONFIGURACIÓN DEL SERVIDOR PARA LA UTILIZACIÓN DEL SERVICIO LDAP.

7.1 INTRODUCCIÓN A SERVICIOS DE DIRECTORIOS.

Un directorio es una base de datos optimizada para lectura, navegación y búsqueda. Los directorios tienden a contener información descriptiva basada en atributos y tienen capacidades de filtrado muy sofisticada. Los directorios generalmente no soportan transacciones complicadas ni esquemas de vuelta atrás como los que se encuentran en los sistemas de bases de datos diseñados para manejar grandes y complejos volúmenes de actualizaciones. Las actualizaciones de los directorios son normalmente cambios simples, o todo o nada, siempre y cuando están permitidos.

Los directorios están afinados para dar una rápida respuesta a grandes volúmenes de búsquedas. Estos tienen la capacidad de replicar la información para incrementar la disponibilidad y la fiabilidad, al tiempo que reducen los tiempos de respuesta. Cuando la información de un directorio se replica, se pueden producir inconsistencias temporales entre las réplicas mientras esta se está sincronizando.

Hay muchas formas diferentes de proveer un servicio de directorio. Diferentes métodos permiten almacenar distintos tipos de información en el directorio, tener distintos requisitos sobre como la información ha de ser referenciada, consultada y actualizada, como es protegida de los accesos no autorizados, etc. Algunos servicios de directorio son locales, es decir, proveen el servicio a un contexto restringido (como por ejemplo, el servicio *finger* en una única máquina). Otros servicios son globales y proveen servicio a un contexto mucho más amplio (como por ejemplo, Internet). Los servicios globales normalmente son distribuidos, esto significa que los datos están repartidos a lo largo de distintos equipos, los cuales cooperan para dar el servicio de directorio. Típicamente, un servicio global define un espacio de nombres uniforme que da la misma visión de los datos, independientemente de donde se está, en relación a los propios datos. El servicio DNS (Domain Name System) es un ejemplo de un sistema de directorio globalmente distribuido.

7.2 LIGHTWEIGHT DIRECTORY ACCESS PROTOCOL

LDAP son las siglas de *Lightweight Directory Access Protocol*. Como su propio nombre indica, es un protocolo ligero para acceder al servicio de directorio, especialmente al basado en X.500. LDAP se ejecuta sobre TCP/IP o sobre otros servicios de transferencia orientado a conexión.

7.2.1 Almacenamiento de la información en el Directorio LDAP

En LDAP, las entradas están organizadas en una estructura jerárquica en árbol. Tradicionalmente, esta estructura reflejaba los límites geográficos y organizacionales. Las entradas que representan países aparecen en la parte superior del árbol. Debajo de ellos, están las entradas que representan los estados y las organizaciones nacionales.

Debajo de éstas, pueden estar las entradas que representan las unidades organizacionales, empleados, impresoras, documentos o todo aquello que pueda imaginarse.

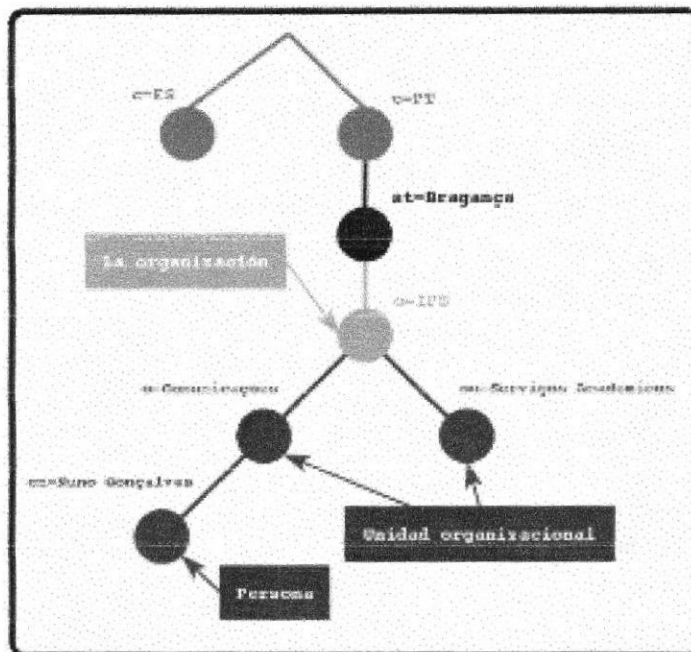


Figura 1. Árbol del directorio LDAP (nombramiento tradicional)

El árbol también se puede organizar basándose en los nombres de dominio de Internet. Este tipo de nombramiento se está volviendo muy popular, ya que permite localizar un servicio de directorio haciendo uso de los DNS.

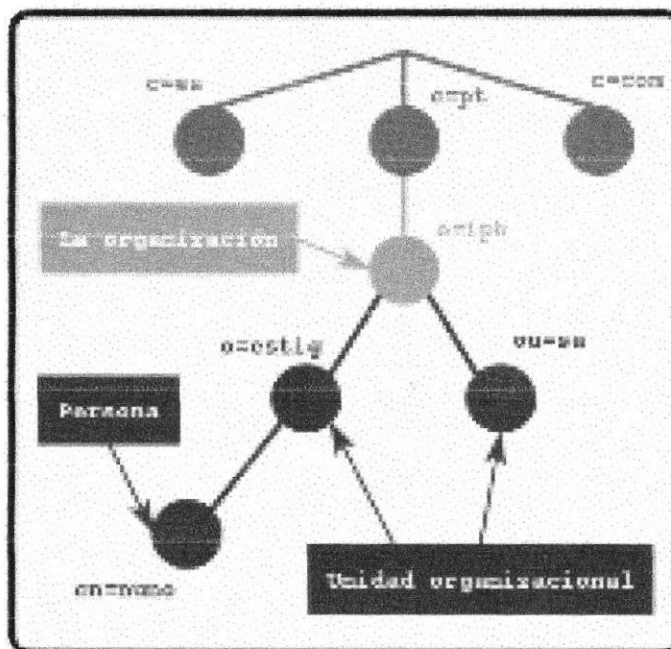


Figura 2. Árbol del directorio LDAP (nombramiento de Internet)

Además, LDAP permite controlar que atributos son requeridos y permitidos en una entrada gracias al uso del atributo denominado *objectClass*. El valor del atributo *objectClass* determina que reglas de diseño (*schema rules*) ha de seguir la entrada.

7.2.2 Referenciación de la información

Una entrada es referenciada por su nombre distinguido, que es construido por el nombre de la propia entrada (llamado *Nombre Relativo Distinguido* o RDN) y la concatenación de los nombres de las entradas que le anteceden. Por ejemplo, la entrada para *Nuno Gonzales* en el ejemplo del nombramiento de Internet anterior tiene el siguiente RDN: *uid=nuno* y su DN sería: *uid=nuno,ou=estig,dc=ipb,dc=pt*.

7.2.3 Acceso a la Información

LDAP define operaciones para interrogar y actualizar el directorio. Provee operaciones para añadir y borrar entradas del directorio, modificar una entrada existente y cambiar el nombre de una entrada. La mayor parte del tiempo, sin embargo, LDAP se utiliza para buscar información almacenada en el directorio. Las operaciones de búsqueda de LDAP permiten buscar entradas que concuerdan con algún criterio especificado por un filtro de búsqueda. La información puede ser solicitada desde cada entrada que concuerda con dicho criterio.

7.2.4 Protección en la información de los accesos no autorizados.

Algunos servicios de directorio no proveen protección, permitiendo a cualquier persona acceder a la información. LDAP provee un mecanismo de autenticación para los clientes, o la confirmación de identidad en un servidor de directorio, facilitando el camino para un control de acceso que proteja la información que el servidor posee. LDAP también soporta los servicios de privacidad e integridad.

7.3 CONSIDERACIONES PREVIAS

La instalación y configuración de OpenLDAP se llevará a cabo de tal manera que al finalizarla, el sistema sobre el que se ha instalado debería estar listo para autenticar usuarios a través del servicio de directorios. Este es el objetivo final de este capítulo, en subsiguientes capítulos se irán añadiendo las funcionalidades necesarias para que cumpla con los requisitos del trabajo.

7.4 INSTALACIÓN DE LOS PAQUETES SLAPD LDAP-UTILS

Para iniciar la configuración de **slapd** se utiliza el comando:

```
# /usr/sbin/dpkg-reconfigure --priority=low slapd
```

(primera parte)

```
# apt-get install slapd ldap-utils
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Se instalará los siguientes paquetes NUEVOS:
  ldap-utils slapd
0 actualizados, 2 se instalarán, 0 para eliminar y 1 no
actualizados.
Se necesita descargar 0B/1042kB de archivos.
Se utilizarán 2884kB de espacio de disco adicional después de
desempaquetar.
Preconfiguring packages ...
```

(segunda parte)

```
find: /var/lib/ldap: No existe el fichero o el directorio

Seleccionando el paquete ldap-utils previamente no
seleccionado.
(Leyendo la base de datos ...)
252690 ficheros y directorios instalados actualmente.)
Desempaquetando ldap-utils (de .../ldap-utils_2.1.30-
1_i386.deb) ...
Seleccionando el paquete slapd previamente no seleccionado.
Desempaquetando slapd (de .../slapd_2.1.30-1_i386.deb) ...
Configurando ldap-utils (2.1.30-1) ...
Configurando slapd (2.1.30-1) ...
Creating initial slapd configuration... done
Creating initial LDAP directory... done
Starting OpenLDAP: slapd.

localepurge: checking system for new locale ...
localepurge: processing locale files ...
localepurge: processing man pages ...
```

7.5 OBSERVACIONES A LA INSTALACIÓN

Una vez más, dependiendo de como se encuentre su sistema y los paquetes que tenga instalados en el mismo, se instalarán y sugerirán más o menos dependencias a la hora de instalar OpenLDAP.

El siguiente bloque muestra la información relativa a los paquetes que se acaban de instalar (dependencias, sugerencias de instalación, etc).

```
$ /usr/bin/apt-cache show slapd ldap-utils
Package: slapd
Priority: optional
Section: net
Installed-Size: 2516
Maintainer: Torsten Landschoff <torsten@debian.org>
Architecture: i386
```



```
Source: openldap2
Version: 2.1.30-1
Provides: ldap-server
Depends: libc6 (>= 2.3.2.ds1-4), libdb4.2, libgcrypt7,
libgnutls10 (>= 1.0.0-0), libgpg-error0 (>= 0.7),
libiodbc2 (>= 3.51.2-2), libldap2 (>= 2.1.17-1),
libltdl3 (>= 1.5.2-2), libsasl2 (>= 2.1.18), libslp1,
libtasn1-2 (>= 0.2.7), libwrap0, zlib1g (>= 1:1.2.1),
debconf (>= 0.5), coreutils (>= 4.5.1-1) | fileutils (>= 4.0i-
1),
psmisc, libldap2 (= 2.1.30-1), perl (>> 5.8.0) | libmime-
base64-perl
Recommends: db4.2-util, libsasl2-modules
Suggests: ldap-utils
Conflicts: umich-ldapd, ldap-server, libbind-dev, bind-dev,
libltdl3 (= 1.5.4-1)
Filename: pool/main/o/openldap2/slapd_2.1.30-1_i386.deb
Size: 940506
MD5sum: 3cc284bf5c0bc8da52a5aabc61c56ca7
Description: OpenLDAP server (slapd)
  This is the OpenLDAP (Lightweight Directory Access Protocol)
  standalone
  server (slapd). The server can be used to provide a standalone
  directory
  service and also includes the slurpd replication server.

Package: ldap-utils
Priority: optional
Section: net
Installed-Size: 328
Maintainer: Torsten Landschoff <torsten@debian.org>
Architecture: i386
Source: openldap2
Version: 2.1.30-1
Replaces: openldap-utils, slapd (<< 2.1.25), openldapd
Provides: ldap-client, openldap-utils
Depends: libc6 (>= 2.3.2.ds1-4), libdb4.2, libgcrypt7,
libgnutls10 (>= 1.0.0-0), libgpg-error0 (>= 0.7),
libiodbc2 (>= 3.51.2-2), libldap2 (>= 2.1.17-1),
libltdl3 (>= 1.5.2-2), libsasl2 (>= 2.1.18), libslp1,
libtasn1-2 (>= 0.2.7), zlib1g (>= 1:1.2.1), libldap2 (= 2.1.30-
1)
Recommends: libsasl2-modules
Conflicts: umich-ldap-utils, openldap-utils, ldap-client
Filename: pool/main/o/openldap2/ldap-utils_2.1.30-1_i386.deb
Size: 113996
MD5sum: 2842fec02d967bbf70943def051d9f58
Description: OpenLDAP utilities
  Utilities from the OpenLDAP (Lightweight Directory Access
  Protocol)
  package. These utilities can access a local or remote LDAP
  server
  and contain all the client programs required to access LDAP
  servers.
```


7.6 COMPROBACIONES INICIALES DE LA INSTALACIÓN

7.6.1 Ejecución del demonio

En este punto, ya se tiene un servidor OpenLDAP instalado y ejecutándose, aunque no está ajustado todavía a los objetivos que persigue este apartado. Para comprobar que efectivamente el demonio **slapd** se está ejecutando, realizaremos un par de consultas al sistema. La primera consiste en ver si el demonio **slapd** se encuentra en la lista de procesos que actualmente se están ejecutando en el sistema:

```
# /bin/ps auxf | /bin/grep slapd
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME
COMMAND
root      4453  0.0  0.5 12144 3004 ?        S    12:52   0:00
/usr/sbin/slapd
root      4455  0.0  0.5 12144 3004 ?        S    12:52   0:00
\_ /usr/sbin/slapd
root      4456  0.0  0.5 12144 3004 ?        S    12:52   0:00
\_ /usr/sbin/slapd
```

La segunda comprobación ha realizar, para ver si el demonio se está realmente ejecutando, es verificar que está escuchando en la red:

```
# /bin/netstat -puta
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
PID/Program name
tcp        0      0 *:ldap                  *:*
LISTEN     4453/slapd
```

7.6.2 Conectando con el servidor

Una vez comprobado que el demonio **slapd** se está ejecutando en el sistema, se verificará que la conexión con el mismo está permitida. Para ello, se realizará una búsqueda sencilla en el directorio. Si todo va bien, se debería mostrar algo similar a:

```
$ /usr/bin/ldapsearch -x -b '' -s base '(objectclass=*)'
namingContexts
# extended LDIF
#
# LDAPv3
# base <> with scope base
# filter: (objectclass=*)
# requesting: namingContexts
#
#
dn:
namingContexts: dc=gsr,dc=pt
# search result
```

```
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
```

Realización de una búsqueda simple con ldapsearch (conexión fallida)

En caso de que la conexión falle, se mostrará el siguiente mensaje:

```
$ ldapsearch -x -b '' -s base '(objectclass=*)' namingContexts
ldap_bind: Can't contact LDAP server (81)
```

