



Manual Programador @visor



CONSEJERÍA DE JUSTICIA Y
ADMINISTRACIÓN PÚBLICA

Documento elaborado por: Francisco José Cantero Villar

Revisado por: Miguel J. Vázquez Rebollo

Versión: 1.0.0

Lugar y fecha: Sevilla, 18 de Agosto de 2005

Contenido:

1	Objetivos	3
2	Integración con terceras aplicaciones	4
3	Ejemplo PL/SQL	8
3.1	Requisitos Previos.....	8
3.2	Demo.....	8
4	Ejemplo Java Web Services.....	13
4.1	Requisitos Previos.....	13
4.2	Instalación de la Demo	13
4.3	Demo.....	14
5	API´s PLSQL	17
5.1	AR_PQ_TYPE.....	17
5.2	AR_PQ_AVISADOR.....	21
6	API Servicios Web.....	30
7	Historia de versiones	31

1 Objetivos

El presente documento tiene como objetivo explicar al usuario desarrollador de aplicaciones ejemplos prácticos de integración tanto con el API PLSQL como los Web Service que ofrece la aplicación.

Como documentación anexa se incluye la especificación de todos los métodos de la API así como el JavaDoc de las clases Java clientes que se pueden generar a partir del descriptor del servicio (WSDL).

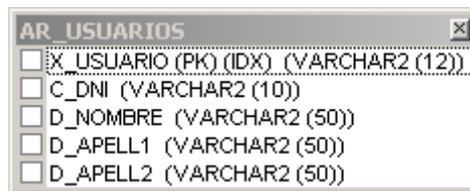
2 Integración con terceras aplicaciones

Cuando confeccionamos un aviso tenemos dos formas de indicar el destinatario, de forma explícita por su dirección de correo electrónico sin más o bien tenemos la posibilidad de indicarlo por el “usuario” o “grupo”.

Si en nuestro sistema tenemos definido una estructura de usuarios y grupos a los cuales estos pertenecen o están suscritos, podemos hacer que el @visorador vea estos datos en tiempo de ejecución. Esto nos permite definir avisos estáticos que se repitan de forma periódica, y los remitentes irán cambiando automáticamente cuando hagamos cambios en los datos de nuestra aplicación.

Para poder conseguirlo debemos definir una serie de vistas que apunten nuestra aplicación que enlace los campos de nuestra aplicación con los que necesita el @visorador. Luego incluimos esta vista a la vista global del @visorador y ya podremos usarla libremente.

La estructura de estas vistas deben corresponderse con las tablas que hay en el propio @visorador, que vemos a continuación:



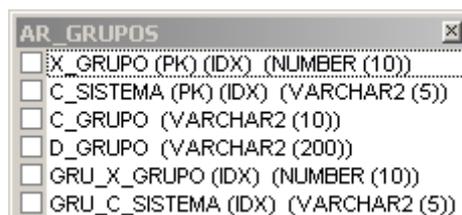
Column Name	Primary Key	Index	Data Type	Length
X_USUARIO	Yes	Yes	VARCHAR2	12
C_DNI	No	No	VARCHAR2	10
D_NOMBRE	No	No	VARCHAR2	50
D_APELL1	No	No	VARCHAR2	50
D_APELL2	No	No	VARCHAR2	50

Figura 1. Tabla de Usuarios



Column Name	Primary Key	Index	Data Type	Length
X_USUARIO_EMAIL	Yes	Yes	NUMBER	10
C_SISTEMA	Yes	Yes	VARCHAR2	5
D_EMAIL	No	No	VARCHAR2	255
USU_X_USUARIO	No	No	VARCHAR2	12

Figura 2. Direcciones de correo válidas de los usuarios.



Column Name	Primary Key	Index	Data Type	Length
X_GRUPO	Yes	Yes	NUMBER	10
C_SISTEMA	Yes	Yes	VARCHAR2	5
C_GRUPO	No	No	VARCHAR2	10
D_GRUPO	No	No	VARCHAR2	200
GRU_X_GRUPO	No	Yes	NUMBER	10
GRU_C_SISTEMA	No	Yes	VARCHAR2	5

Figura 3. Grupos a los que se suscriben/pertenecen los usuarios (tabla anidada sobre si

misma).



Figura 4. Usuarios suscritos/pertenecen a un grupo.

Veamos para el caso de [port@firmas](#) un ejemplo de integración, de forma que al usuario Oracle del @visorador se le ha concedido SELECT sobre las tablas de usuarios y grupos con GRANT OPTION para que estos permisos se puedan trasladar al rol. Con estas premisas las vistas deberían ser de la siguiente forma:

AR_V_PF_USUARIOS

```
CREATE OR REPLACE FORCE VIEW AVISADOR.AR_V_PF_USUARIOS
(X_USUARIO, C_DNI, D_NOMBRE, D_APELL1, D_APELL2)
AS
SELECT
AR_PQ_BASE64.FU_ENCODE_CHAR(USU.C_DNI) X_USUARIO,
USU.C_DNI C_DNI,
USU.D_NOMBRE D_NOMBRE,
USU.D_APELL1 D_APELL1,
USU.D_APELL2 D_APELL2
FROM PF_USUARIOS USU;
```

Se recomienda que la vista empiece por las iniciales de la aplicación @visorador (AR), seguido de un descriptor del objeto, en este caso V de “vista”, y seguido las iniciales de la aplicación (Sistema) a la que vamos a acceder, en el caso de ejemplo de [Port@firmas](#), (PF).

Como clave primaria usaremos la conversión a Base64 del DNI para enmascararlos y así esta información no aparezca tantas veces replicada en la base de datos.

Es importante que nuestra vista de usuarios se limite a DNI’s no duplicados, válidos y con un formato 99999999X y mayúsculas, sin guiones, espacios u otro tipo de separador. Dentro del paquete AR_PQ_TOOL existen varias funciones para ayudarnos a depurar los datos de entrada.

El resto de vistas deben seguir como siguen:

AR_V_PF_USUARIOS_EMAIL

```
CREATE OR REPLACE FORCE VIEW AVISADOR.AR_V_PF_USUARIOS_EMAIL
(X_USUARIO_EMAIL, C_SISTEMA, D_EMAIL, USU_X_USUARIO)
AS
SELECT
A.X_USUARIO_EMAIL,
A.C_SISTEMA,
A.D_EMAIL,
A.USU_X_USUARIO
FROM
```

```
(SELECT
M.X_USUARIO_EMAIL,
'PF' C_SISTEMA,
M.D_EMAIL,
AR_PQ_BASE64.FU_ENCODE_CHAR(U.C_DNI) USU_X_USUARIO
FROM
PF_USUARIOS U,
PF_USUARIOS_EMAIL M
WHERE U.X_USUARIO=M.USU_X_USUARIO AND M.L_NOTIFICAR LIKE 'S') A;
```

AR_V_PF_GRUPOS

```
CREATE OR REPLACE FORCE VIEW AVISADOR.AR_V_PF_GRUPOS
(X_GRUPO, C_SISTEMA, C_GRUPO, D_GRUPO, GRU_X_GRUPO,
GRU_C_SISTEMA)
AS
SELECT
1 X_GRUPO,
'PF' C_SISTEMA,
'PORT@FIRMA' C_GRUPO,
'Grupo de usuarios Port@firma' D_GRUPO,
1 GRU_X_GRUPO,
" GRU_C_SISTEMA
FROM DUAL;
```

En este caso, la aplicación no tiene una tabla de grupos, pero creamos uno para suscribir todos los usuarios de [Port@firmas](#) a un grupo a tal efecto.

AR_V_PF_SUBSCRIPCIONES

```
CREATE OR REPLACE FORCE VIEW AVISADOR.AR_V_PF_SUBSCRIPCIONES
(X_SUBSCRIPCION, C_SISTEMA, USU_X_USUARIO, GRU_X_GRUPO, GRU_C_SISTEMA)
AS
SELECT
ROWNUM X_SUBSCRIPCION,
'PF' C_SISTEMA,
X_USUARIO USU_X_USUARIO,
X_GRUPO GRU_X_GRUPO,
C_SISTEMA GRU_C_SISTEMA
FROM AR_V_PF_USUARIOS, AR_V_PF_GRUPOS;
```

Una vez que tenemos las vistas las unimos a las vistas principales sobre las que consulta el @visorador a la hora de generar los mensajes de la siguiente forma:

AR_V_USUARIOS

```
CREATE OR REPLACE FORCE VIEW AVISADOR.AR_V_USUARIOS
(X_USUARIO, C_DNI, D_NOMBRE, D_APELL1, D_APELL2)
AS
SELECT
A.X_USUARIO,
A.C_DNI,
A.D_NOMBRE,
A.D_APELL1,
A.D_APELL2
FROM
(SELECT X_USUARIO,C_DNI,D_NOMBRE,D_APELL1,D_APELL2 FROM AR_V_SV_USUARIOS
UNION
SELECT X_USUARIO,C_DNI,D_NOMBRE,D_APELL1,D_APELL2 FROM AR_V_PF_USUARIOS
UNION
SELECT X_USUARIO,C_DNI,D_NOMBRE,D_APELL1,D_APELL2 FROM AR_V_AV_USUARIOS
UNION
SELECT X_USUARIO,C_DNI,D_NOMBRE,D_APELL1,D_APELL2 FROM AR_V_ST_USUARIOS
UNION
SELECT X_USUARIO,C_DNI,D_NOMBRE,D_APELL1,D_APELL2 FROM AR_V_AR_USUARIOS) A;
```

AR_V_USUARIOS_EMAIL

```
CREATE OR REPLACE FORCE VIEW AVISADOR.AR_V_USUARIOS_EMAIL
(X_USUARIO_EMAIL, C_SISTEMA, D_EMAIL, USU_X_USUARIO)
AS
SELECT
```

```
A.X_USUARIO_EMAIL,  
A.C_SISTEMA,  
A.D_EMAIL,  
A.USU_X_USUARIO  
FROM  
(SELECT X_USUARIO_EMAIL,C_SISTEMA,D_EMAIL,USU_X_USUARIO FROM AR_V_SV_USUARIOS_EMAIL  
UNION  
SELECT X_USUARIO_EMAIL,C_SISTEMA,D_EMAIL,USU_X_USUARIO FROM AR_V_PF_USUARIOS_EMAIL  
UNION  
SELECT X_USUARIO_EMAIL,C_SISTEMA,D_EMAIL,USU_X_USUARIO FROM AR_V_AV_USUARIOS_EMAIL  
UNION  
SELECT X_USUARIO_EMAIL,C_SISTEMA,D_EMAIL,USU_X_USUARIO FROM AR_V_ST_USUARIOS_EMAIL  
UNION  
SELECT X_USUARIO_EMAIL,C_SISTEMA,D_EMAIL,USU_X_USUARIO FROM AR_V_AR_USUARIOS_EMAIL) A;
```

AR_V_GRUPOS

```
CREATE OR REPLACE FORCE VIEW AVISADOR.AR_V_GRUPOS  
(X_GRUPO, C_SISTEMA, C_GRUPO, D_GRUPO, GRU_X_GRUPO,  
GRU_C_SISTEMA)  
AS  
SELECT  
A.X_GRUPO,  
A.C_SISTEMA,  
A.C_GRUPO,  
A.D_GRUPO,  
A.GRU_X_GRUPO,  
A.GRU_C_SISTEMA  
FROM  
(SELECT X_GRUPO,C_SISTEMA,C_GRUPO,D_GRUPO,GRU_X_GRUPO,GRU_C_SISTEMA FROM AR_V_SV_GRUPOS  
UNION  
SELECT X_GRUPO,C_SISTEMA,C_GRUPO,D_GRUPO,GRU_X_GRUPO,GRU_C_SISTEMA FROM AR_V_PF_GRUPOS  
UNION  
SELECT X_GRUPO,C_SISTEMA,C_GRUPO,D_GRUPO,GRU_X_GRUPO,GRU_C_SISTEMA FROM AR_V_AV_GRUPOS  
UNION  
SELECT X_GRUPO,C_SISTEMA,C_GRUPO,D_GRUPO,GRU_X_GRUPO,GRU_C_SISTEMA FROM AR_V_ST_GRUPOS  
UNION  
SELECT X_GRUPO,C_SISTEMA,C_GRUPO,D_GRUPO,GRU_X_GRUPO,GRU_C_SISTEMA FROM AR_V_AR_GRUPOS) A;
```

AR_V_SUBSCRIPCIONES

```
CREATE OR REPLACE FORCE VIEW AVISADOR.AR_V_SUBSCRIPCIONES  
(X_SUBSCRIPCION, C_SISTEMA, USU_X_USUARIO, GRU_X_GRUPO, GRU_C_SISTEMA)  
AS  
SELECT  
A.X_SUBSCRIPCION,  
A.C_SISTEMA,  
A.USU_X_USUARIO,  
A.GRU_X_GRUPO,  
A.GRU_C_SISTEMA  
FROM  
(SELECT X_SUBSCRIPCION,C_SISTEMA,USU_X_USUARIO,GRU_X_GRUPO,GRU_C_SISTEMA FROM  
AR_V_PF_SUBSCRIPCIONES  
UNION  
SELECT X_SUBSCRIPCION,C_SISTEMA,USU_X_USUARIO,GRU_X_GRUPO,GRU_C_SISTEMA FROM  
AR_V_SV_SUBSCRIPCIONES  
UNION  
SELECT X_SUBSCRIPCION,C_SISTEMA,USU_X_USUARIO,GRU_X_GRUPO,GRU_C_SISTEMA FROM AR_V_AV_SUBSCRIPCIONES  
UNION  
SELECT X_SUBSCRIPCION,C_SISTEMA,USU_X_USUARIO,GRU_X_GRUPO,GRU_C_SISTEMA FROM  
AR_V_ST_SUBSCRIPCIONES) A;
```

No importa el orden el que se incluya dado que las uniones implican una eliminación de duplicados y ordenación implícita de elementos.

3 Ejemplo PL/SQL

3.1 Requisitos Previos

Necesitamos cumplir los siguientes requisitos:

- Acceso a la base de datos donde esté situado el [@visorador](#) y un usuario con los permisos y sinónimos adecuados para poder llamar a las API's de [@visorador](#).
- Cliente PL/SQL tipo TOAD, TORA o SQLPLUS.

3.2 Demo

El ejemplo se trata de un program-unit sencillo que llama a las principales rutinas de la API a modo de ejemplo explicando cada una de ellas.

```
DECLARE
AVISO_C_HASH VARCHAR2(10);
AVISOS COL_AR_AVISOS;
I NUMBER;
J NUMBER;
K NUMBER;
AVISO OBJ_AR_AVISOS;
USUARIOS COL_AR_USUARIOS;
USUARIOS_DESTINATARIO COL_AR_USUARIOS_DESTINATARIO;
GRUPOS COL_AR_GRUPOS;
GRUPOS_DESTINATARIO COL_AR_GRUPOS_DESTINATARIO;
MENSAJES COL_AR_MENSAJES;
ENVIOS COL_AR_ENVIOS_EMAIL;
BEGIN
    -- INICIALIZACIÓN DE VARIABLES
    AVISOS := COL_AR_AVISOS();
    I := 1;
    J := 1;
    AVISO := OBJ_AR_AVISOS(NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL);
    USUARIOS := COL_AR_USUARIOS();
    USUARIOS_DESTINATARIO := COL_AR_USUARIOS_DESTINATARIO();
    GRUPOS := COL_AR_GRUPOS();
    GRUPOS_DESTINATARIO := COL_AR_GRUPOS_DESTINATARIO();
    MENSAJES := COL_AR_MENSAJES();
    ENVIOS := COL_AR_ENVIOS_EMAIL();

    -- LO PRIMERO QUE HAY QUE HACER ES CREAR EL AVISO
    -- PARA ELLO TENEMOS DOS OPCIONES:
    -- PASARLE COMO PARÁMETRO UN OBJETO CON LOS CAMPOS RELLENOS
    -- PASARLE COMO PARÁMETRO TODOS LOS CAMPOS QUE SE PUEDEN MODIFICAR DE
    MANERA SIMPLE
    -- VAMOS A UTILIZAR LA SEGUNDA OPCIÓN Y SOLO VAMOS A RELLENAR EL TEXTO
    -- LOS DEMÁS CAMPOS QUE SE PUEDEN MODIFICAR SE RELLENARÁN CON VALORES POR DEFECTO
    -- ESTOS VALORES POR DEFECTO SON:
    -- N_PERIODICIDAD: 0 (NO TIENE PERIODICIDAD, ES DECIR, SOLO SE ENVIARÁ 1 VEZ)
    -- C_TIPO:M (MÚLTIPLE)
    -- N_PRIORIDAD:3 (PRIORIDAD MEDIA)
    AVISO_C_HASH := AR_PQ_AVISADOR.FU_AVISO_NUEVO('Texto inicial');

    -- HACEMOS UNA CONSULTA PARA MOSTRAR LO QUE HA CREADO EL PROCEDIMIENTO ANTERIOR
    AR_PQ_AVISADOR.PR_AVISO_CONSULTA(AVISOS,AVISO_C_HASH);

    -- VEMOS EL AVISO QUE SE HA CREADO CON LOS VALORES POR DEFECTO
    DBMS_OUTPUT.PUT_LINE('AVISO CREADO POR DEFECTO');
    DBMS_OUTPUT.PUT_LINE('-----');
    DBMS_OUTPUT.PUT('X_AVISO: || AVISOS(I).X_AVISO);
    DBMS_OUTPUT.PUT(' C_HASH: || AVISOS(I).C_HASH);
    DBMS_OUTPUT.PUT(' F_AVISO: || AVISOS(I).F_AVISO);
```

```

DBMS_OUTPUT.PUT(' N_PERIODICIDAD:' || AVISOS(I).N_PERIODICIDAD);
DBMS_OUTPUT.PUT(' C_TIPO:' || AVISOS(I).C_TIPO);
DBMS_OUTPUT.PUT(' EST_X_ESTADO:' || AVISOS(I).EST_X_ESTADO);
DBMS_OUTPUT.PUT(' F_ESTADO:' || AVISOS(I).F_ESTADO);
DBMS_OUTPUT.PUT(' T_OBSERVACIONES:' || AVISOS(I).T_OBSERVACIONES);
DBMS_OUTPUT.PUT(' N_PRIORIDAD:' || AVISOS(I).N_PRIORIDAD);
DBMS_OUTPUT.PUT_LINE("");
DBMS_OUTPUT.PUT_LINE("");

-- AHORA VAMOS A MODIFICAR ALGUNOS CAMPOS QUE ANTES HEMOS CREADO POR DEFECTO
-- AL IGUAL QUE LA CREACIÓN TENEMOS DOS OPCIONES:
-- CAMPO A CAMPO SIMPLE
-- UN OBJETO CON LOS CAMPOS QUE QUEREMOS ACTUALIZAR
-- ESTA VEZ SE VA A UTILIZAR EL OBJETO PARA IR VARIANDO EN EL EJEMPLO
AVISO.C_HASH := AVISO.C_HASH;
AVISO.T_AVISO := 'Cambio de texto';
AVISO.N_PERIODICIDAD := 36000; -- SE VA A REENVIAR CADA HORA
AVISO.N_VECES:=3;
AVISO.C_TIPO := 'U'; -- UNITARIO
AVISO.N_PRIORIDAD := 5; -- PRIORIDAD ALTA
AVISO.D_ASUNTO:='Prueba';
-- UNA VEZ RELLENO EL OBJETO HACEMOS LA MODIFICACIÓN
AR_PQ_AVISADOR.PR_AVISO_MODIFICA(AVISO);

-- HACEMOS UNA CONSULTA PARA MOSTRAR LO QUE HA CREADO EL PROCEDIMIENTO ANTERIOR
AR_PQ_AVISADOR.PR_AVISO_CONSULTA(AVISOS,AVISO.C_HASH);

-- VEMOS EL AVISO QUE SE HA CREADO CON LOS VALORES POR DEFECTO
DBMS_OUTPUT.PUT_LINE('AVISO MODIFICADO');
DBMS_OUTPUT.PUT_LINE('-----');
DBMS_OUTPUT.PUT('X_AVISO:' || AVISOS(I).X_AVISO);
DBMS_OUTPUT.PUT(' C_HASH:' || AVISOS(I).C_HASH);
DBMS_OUTPUT.PUT(' F_AVISO:' || AVISOS(I).F_AVISO);
DBMS_OUTPUT.PUT(' N_PERIODICIDAD:' || AVISOS(I).N_PERIODICIDAD);
DBMS_OUTPUT.PUT(' C_TIPO:' || AVISOS(I).C_TIPO);
DBMS_OUTPUT.PUT(' EST_X_ESTADO:' || AVISOS(I).EST_X_ESTADO);
DBMS_OUTPUT.PUT(' F_ESTADO:' || AVISOS(I).F_ESTADO);
DBMS_OUTPUT.PUT(' T_OBSERVACIONES:' || AVISOS(I).T_OBSERVACIONES);
DBMS_OUTPUT.PUT(' N_PRIORIDAD:' || AVISOS(I).N_PRIORIDAD);
DBMS_OUTPUT.PUT_LINE("");
DBMS_OUTPUT.PUT_LINE("");

-- AHORA VAMOS A AÑADIR VARIOS USUARIOS DESTINATARIOS
-- CONSULTAMOS LA VISTA DE USUARIOS Y AÑADIMOS POR EJEMPLO LOS 3 PRIMEROS
AR_PQ_AVISADOR.PR_USUARIOS_CONSULTA(USUARIOS,NULL);

-- PARA AÑADIR USUARIOS TENEMOS 3 OPCIONES Y VAMOS A HACER USO DE LAS 2 PRIMERAS:
-- AÑADIRLO POR DNI:
-- AÑADIRLO POR X_USUARIO:
-- AÑADIRLO POR CORREO:
AR_PQ_AVISADOR.PR_AVISO_ANADE_USUARIO(USUARIOS(1).C_DNI,AVISO.C_HASH);
AR_PQ_AVISADOR.PR_AVISO_ANADE_USUARIO(USUARIOS(2).C_DNI,AVISO.C_HASH);
AR_PQ_AVISADOR.PR_AVISO_ANADE_USUARIO(USUARIOS(3).X_USUARIO,AVISO.C_HASH);

-- AHORA TENEMOS 3 USUARIOS DESTINATARIOS
-- CONSULTAMOS LA TABLA DE USUARIOS DESTINATARIOS PARA COMPROBARLO:
AR_PQ_AVISADOR.PR_AVISO_CONSULTA_USUARIOS(USUARIOS_DESTINATARIO,AVISO.C_HASH);

-- SE MUESTRA
DBMS_OUTPUT.PUT_LINE('TABLA DE USUARIOS DESTINATARIO CON 3 USUARIOS INSERTADOS');
DBMS_OUTPUT.PUT_LINE('-----');
FOR J IN 1 .. USUARIOS_DESTINATARIO.COUNT LOOP

DBMS_OUTPUT.PUT('X_USUARIO_DESTINATARIO=||USUARIOS_DESTINATARIO(J).X_USUARIO_DESTINATARIO);
DBMS_OUTPUT.PUT('AVS_X_AVISO=||USUARIOS_DESTINATARIO(J).AVS_X_AVISO);
DBMS_OUTPUT.PUT('USU_X_USUARIO=||USUARIOS_DESTINATARIO(J).USU_X_USUARIO);
DBMS_OUTPUT.PUT_LINE("");

END LOOP;
DBMS_OUTPUT.PUT_LINE("");
DBMS_OUTPUT.PUT_LINE("");

-- AHORA VAMOS A ELIMINAR UN USUARIO, POR EJEMPLO EL PRIMERO
AR_PQ_AVISADOR.PR_AVISO_ELIMINA_USUARIO(USUARIOS_DESTINATARIO(1).USU_X_USUARIO,AVISO.C_HASH);

-- CONSULTAMOS LA TABLA DE USUARIOS DESTINATARIOS PARA COMPROBARLO:
AR_PQ_AVISADOR.PR_AVISO_CONSULTA_USUARIOS(USUARIOS_DESTINATARIO,AVISO.C_HASH);

-- Y SE MUESTRA DE NUEVO ESTA TABLA
DBMS_OUTPUT.PUT_LINE('TABLA DE USUARIOS DESTINATARIO DESPUES DEL BORRADO DEL PRIMERO');
DBMS_OUTPUT.PUT_LINE('-----');
FOR J IN 1 .. USUARIOS_DESTINATARIO.COUNT LOOP

```

```
DBMS_OUTPUT.PUT('X_USUARIO_DESTINATARIO=||USUARIOS_DESTINATARIO(J).X_USUARIO_DESTINATARIO);
      DBMS_OUTPUT.PUT('AVS_X_Aviso=||USUARIOS_DESTINATARIO(J).AVS_X_Aviso);
      DBMS_OUTPUT.PUT('USU_X_USUARIO=||USUARIOS_DESTINATARIO(J).USU_X_USUARIO);
      DBMS_OUTPUT.PUT_LINE("");
END LOOP;
DBMS_OUTPUT.PUT_LINE("");
DBMS_OUTPUT.PUT_LINE("");

-- AHORA VAMOS A HACER EL MISMO PROCESO PERO CON GRUPOS

-- CONSULTAMOS LA VISTA DE GRUPOS Y AÑADIMOS POR EJEMPLO LOS 3 PRIMEROS
AR_PQ_AVISADOR.PR_GRUPOS_CONSULTA(GRUPOS,NULL);

-- PARA AÑADIR GRUPOS TENEMOS 2 OPCIONES Y VAMOS A HACER USO DE LAS 2
-- AÑADIRLO POR C_GRUPO:
-- AÑADIRLO POR X_GRUPO Y C_SISTEMA:
AR_PQ_AVISADOR.PR_Aviso_ANADE_GRUPO(GRUPOS(1).C_GRUPO,AVISO_C_HASH);
AR_PQ_AVISADOR.PR_Aviso_ANADE_GRUPO(GRUPOS(2).C_GRUPO,AVISO_C_HASH);
AR_PQ_AVISADOR.PR_Aviso_ANADE_GRUPO(GRUPOS(3).X_GRUPO,GRUPOS(3).C_SISTEMA,AVISO_C_HASH);

-- AHORA TENEMOS 3 GRUPOS DESTINATARIOS TAMBIÉN
-- CONSULTAMOS LA TABLA DE GRUPOS DESTINATARIOS PARA COMPROBARLO:
AR_PQ_AVISADOR.PR_Aviso_CONSULTA_GRUPOS(GRUPOS_DESTINATARIO,AVISO_C_HASH);

-- SE MUESTRA
DBMS_OUTPUT.PUT_LINE('TABLA DE GRUPOS DESTINATARIO CON 3 GRUPOS INSERTADOS');
DBMS_OUTPUT.PUT_LINE('-----');
FOR J IN 1 .. GRUPOS_DESTINATARIO.COUNT LOOP

DBMS_OUTPUT.PUT('X_GRUPO_DESTINATARIO=||GRUPOS_DESTINATARIO(J).X_GRUPO_DESTINATARIO);
      DBMS_OUTPUT.PUT('AVS_X_Aviso=||GRUPOS_DESTINATARIO(J).AVS_X_Aviso);
      DBMS_OUTPUT.PUT('GRU_X_GRUPO=||GRUPOS_DESTINATARIO(J).GRU_X_GRUPO);
      DBMS_OUTPUT.PUT('GRU_C_SISTEMA=||GRUPOS_DESTINATARIO(J).GRU_C_SISTEMA);
      DBMS_OUTPUT.PUT_LINE("");
END LOOP;
DBMS_OUTPUT.PUT_LINE("");
DBMS_OUTPUT.PUT_LINE("");

-- AHORA VAMOS A ELIMINAR UN GRUPO, POR EJEMPLO EL TERCERO

AR_PQ_AVISADOR.PR_Aviso_ELIMINA_GRUPO(GRUPOS_DESTINATARIO(3).GRU_X_GRUPO,GRUPOS_DESTINATARIO(3).GRU_C_SISTEMA,AVISO_C_HASH);

-- CONSULTAMOS LA TABLA DE USUARIOS DESTINATARIOS PARA COMPROBARLO:
AR_PQ_AVISADOR.PR_Aviso_CONSULTA_GRUPOS(GRUPOS_DESTINATARIO,AVISO_C_HASH);

-- SE MUESTRA DE NUEVO LA TABLA
DBMS_OUTPUT.PUT_LINE('TABLA DE GRUPOS DESTINATARIO DESPUES DEL BORRADO DEL TERCERO');
DBMS_OUTPUT.PUT_LINE('-----');
FOR J IN 1 .. GRUPOS_DESTINATARIO.COUNT LOOP

DBMS_OUTPUT.PUT('X_GRUPO_DESTINATARIO=||GRUPOS_DESTINATARIO(J).X_GRUPO_DESTINATARIO);
      DBMS_OUTPUT.PUT('AVS_X_Aviso=||GRUPOS_DESTINATARIO(J).AVS_X_Aviso);
      DBMS_OUTPUT.PUT('GRU_X_GRUPO=||GRUPOS_DESTINATARIO(J).GRU_X_GRUPO);
      DBMS_OUTPUT.PUT('GRU_C_SISTEMA=||GRUPOS_DESTINATARIO(J).GRU_C_SISTEMA);
      DBMS_OUTPUT.PUT_LINE("");
END LOOP;
DBMS_OUTPUT.PUT_LINE("");
DBMS_OUTPUT.PUT_LINE("");

-- DESPUÉS DE TODAS ESTAS OPERACIONES TENEMOS 2 USUARIOS Y 2 GRUPOS DESTINATARIOS
-- EN ESTE CASO YA PODEMOS ACTIVAR EL MENSAJE
-- SI NO SE TUVIERA NINGÚN DESTINATARIO ENTONCES NO SE ACTIVARÍA PERO ESTE NO ES NUESTRO CASO
-- COMPROBAMOS QUE EL ESTADO DEL AVISO ES REDACCIÓN
DBMS_OUTPUT.PUT_LINE('ESTADO DEL AVISO ANTES DE LA ACTIVACIÓN:');
AR_PQ_AVISADOR.FU_Aviso_ESTADO_CONSULTA(AVISO_C_HASH);
DBMS_OUTPUT.PUT_LINE("");
DBMS_OUTPUT.PUT_LINE("");

-- VIENDO QUE EL AVISO ESTÁ EN REDACCIÓN SEGUIMOS Y LO ACTIVAMOS
AR_PQ_AVISADOR.PR_Aviso_ESTADO_ACTIVACION(AVISO_C_HASH);

-- OBSERVAMOS EL CAMBIO DE ESTADO
DBMS_OUTPUT.PUT_LINE('ESTADO DEL AVISO DESPUÉS DE LA ACTIVACIÓN:');
AR_PQ_AVISADOR.FU_Aviso_ESTADO_CONSULTA(AVISO_C_HASH);
DBMS_OUTPUT.PUT_LINE("");

-- HASTA AQUI TODO LO REFERENTE A LA ADMINISTRACIÓN DEL AVISO
```

```

-- ESTE AVISO SE ENCUENTRA AHORA EN ESPERA DE QUE EL DEMONIO CREE LOS MENSAJES PERTINENTES
-- Y LOS ENVÍE A LOS DESTINATARIOS ESPECIFICADOS
-- ENTONCES PASADO UN TIEMPO PODREMOS VER TODOS LOS MENSAJES GENERADOS POR EL DEMONIO Y
-- LOS ESTADOS DE CADA ENVÍO
-- PARA VER TODOS LOS MENSAJES GENERADOS HACEMOS UNA CONSULTA A LA TABLA MENSAJES
-- AHORA TENEMOS 3 USUARIOS DESTINATARIOS

/*-- CONSULTAMOS LA TABLA DE MENSAJES DE ESTE AVISO
AR_PQ_AVISADOR.PR_MENSAJES_CONSULTA(MENSAJES,AVISO_C_HASH);

-- SE MUESTRA
DBMS_OUTPUT.PUT_LINE('TABLA DE MENSAJES');
DBMS_OUTPUT.PUT_LINE('-----');
FOR J IN 1 .. MENSAJES.COUNT LOOP
  DBMS_OUTPUT.PUT('X_MENSAJE='||MENSAJES(J).X_MENSAJE);
  DBMS_OUTPUT.PUT('C_HASH='||MENSAJES(J).C_HASH);
  DBMS_OUTPUT.PUT('AVS_X_AVIOS='||MENSAJES(J).AVS_X_AVIOS);
  DBMS_OUTPUT.PUT('F_MENSAJE='||MENSAJES(J).F_MENSAJE);
  DBMS_OUTPUT.PUT('T_MENSAJE='||MENSAJES(J).T_MENSAJE);
  DBMS_OUTPUT.PUT_LINE('');

  -- PARA CADA MENSAJE SE MIRA EL ESTADO DEL ENVIO PARA CADA DESTINATARIO
  AR_PQ_AVISADOR.PR_ENVIOS_EMAIL_CONSULTA(ENVIOS,MENSAJES(J).C_HASH);

  DBMS_OUTPUT.PUT_LINE('          TABLA DE ENVIOS');
  DBMS_OUTPUT.PUT_LINE('          -----');
  FOR K IN 1 .. ENVIOS.COUNT LOOP
    DBMS_OUTPUT.PUT('          X_ENVIO_EMAIL='||ENVIOS(K).X_ENVIO_EMAIL);
    DBMS_OUTPUT.PUT('          MEN_X_MENSAJE='||ENVIOS(K).MEN_X_MENSAJE);
    DBMS_OUTPUT.PUT('          D_EMAIL='||ENVIOS(K).D_EMAIL);
    DBMS_OUTPUT.PUT('          EST_X_ESTADO='||ENVIOS(K).EST_X_ESTADO);
    DBMS_OUTPUT.PUT('          F_ESTADO='||ENVIOS(K).F_ESTADO);
    DBMS_OUTPUT.PUT('          T_OBSERVACIONES='||ENVIOS(K).T_OBSERVACIONES);
    DBMS_OUTPUT.PUT_LINE('');
  END LOOP;
  DBMS_OUTPUT.PUT_LINE('');
END LOOP;
DBMS_OUTPUT.PUT_LINE('');
DBMS_OUTPUT.PUT_LINE('');
*/

END;

```

Si activamos la salida estandar de la consola nos saldrá el siguiente log como se muestra en la siguiente pantalla.

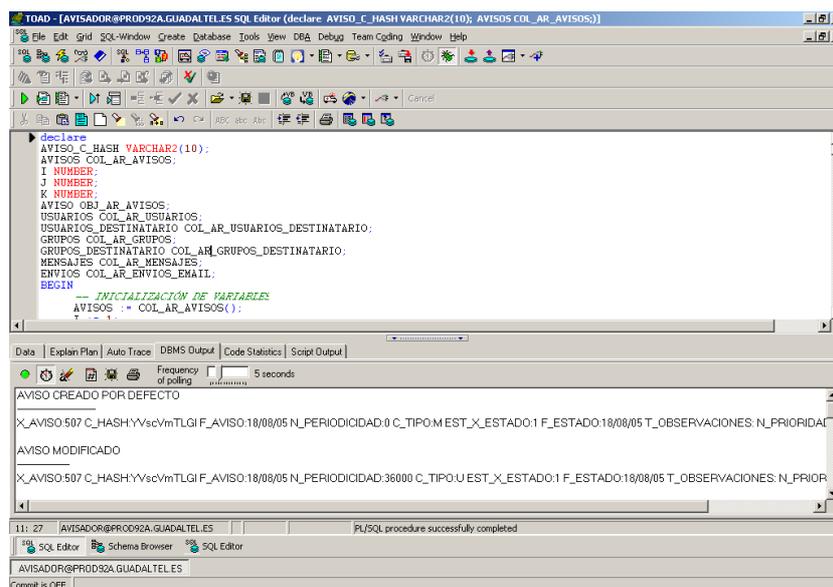


Figura 5. Ejemplo PL/SQL

El log completo de este ejemplo:

AVISO CREADO POR DEFECTO

X_AVISO:507 C_HASH:YVscVmTLGI F_AVISO:18/08/05 N_PERIODICIDAD:0 C_TIPO:M EST_X_ESTADO:1 F_ESTADO:18/08/05
T_OBSERVACIONES: N_PRIORIDAD:3

AVISO MODIFICADO

X_AVISO:507 C_HASH:YVscVmTLGI F_AVISO:18/08/05 N_PERIODICIDAD:36000 C_TIPO:U EST_X_ESTADO:1 F_ESTADO:18/08/05
T_OBSERVACIONES: N_PRIORIDAD:5

TABLA DE USUARIOS DESTINATARIO CON 3 USUARIOS INSERTADOS

X_USUARIO_DESTINATARIO=141AVS_X_AVISO=507USU_X_USUARIO=NzU4MDk0ODNC
X_USUARIO_DESTINATARIO=139AVS_X_AVISO=507USU_X_USUARIO=Nzc4MDYwMzBD
X_USUARIO_DESTINATARIO=140AVS_X_AVISO=507USU_X_USUARIO=NDg4NTY2NjZD

TABLA DE USUARIOS DESTINATARIO DESPUES DEL BORRADO DEL PRIMERO

X_USUARIO_DESTINATARIO=139AVS_X_AVISO=507USU_X_USUARIO=Nzc4MDYwMzBD
X_USUARIO_DESTINATARIO=140AVS_X_AVISO=507USU_X_USUARIO=NDg4NTY2NjZD

TABLA DE GRUPOS DESTINATARIO CON 3 GRUPOS INSERTADOS

X_GRUPO_DESTINATARIO=108AVS_X_AVISO=507GRU_X_GRUPO=215GRU_C_SISTEMA=AV
X_GRUPO_DESTINATARIO=106AVS_X_AVISO=507GRU_X_GRUPO=206GRU_C_SISTEMA=AV
X_GRUPO_DESTINATARIO=107AVS_X_AVISO=507GRU_X_GRUPO=555GRU_C_SISTEMA=AV

TABLA DE GRUPOS DESTINATARIO DESPUES DEL BORRADO DEL TERCERO

X_GRUPO_DESTINATARIO=108AVS_X_AVISO=507GRU_X_GRUPO=215GRU_C_SISTEMA=AV
X_GRUPO_DESTINATARIO=106AVS_X_AVISO=507GRU_X_GRUPO=206GRU_C_SISTEMA=AV

ESTADO DEL AVISO ANTES DE LA ACTIVACIÓN:REDACCION

ESTADO DEL AVISO DESPUÉS DE LA ACTIVACIÓN:EN ESPERA

4 Ejemplo Java Web Services

4.1 Requisitos Previos

Para poder ejecutar la web de ejemplo necesitamos cumplir los siguientes requisitos:

- Servidor de aplicaciones Tomcat o similar para poder desplegar el WAR de ejemplo.
- Tener ejecutando [@visorador](#) en este u otro servidor de aplicaciones al que tengamos acceso.

4.2 Instalación de la Demo

Dentro del Kit @visorador se ha incluido una aplicación web de ejemplo con un cliente Web Services generado a partir del WSDL del servicio. Dentro del Kit @visorador dentro de la carpeta de Ejemplos, Webservice y localizamos el WAR.

Para instalar el WAR de la aplicación debemos entrar en la herramienta de administración de nuestro Tomcat.

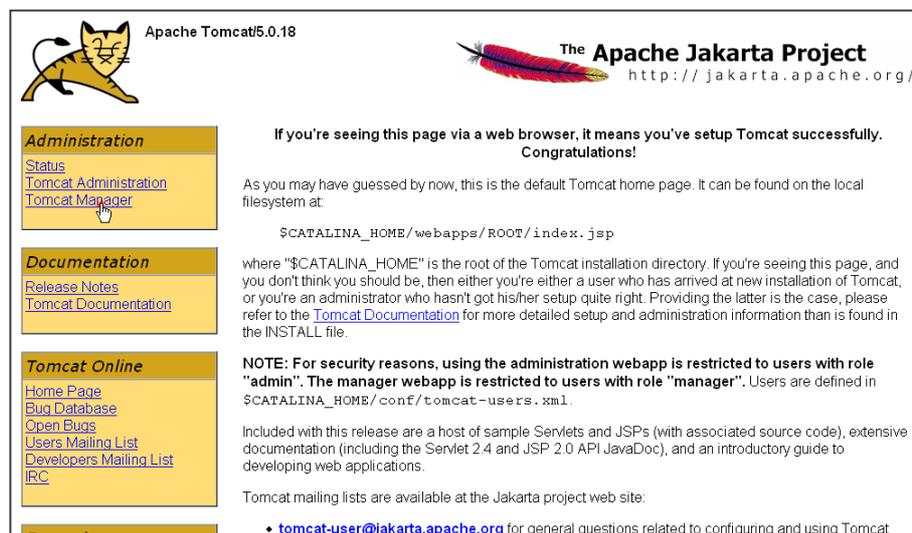


Figura 6. Pantalla de bienvenida de Tomcat

Entramos en el Tomcat Manager, nos vamos al apartado de despliegue de nuevas aplicaciones, le damos al botón “Examinar” y buscamos el War dentro de la carpeta War Ejemplo y lo incorporamos al Tomcat.

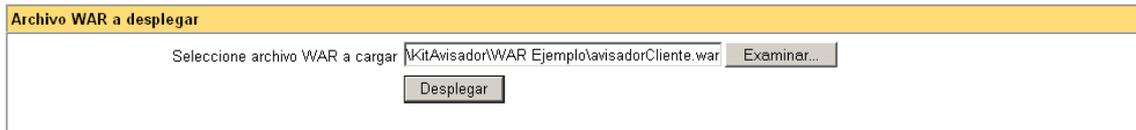


Figura 7. Despliegue de la aplicación en el Tomcat.

Si la aplicación se ha desplegado correctamente nos aparecerá ahora junto al resto de aplicaciones desplegadas.

/avisadorcliente	Cliente WS Avisador	true	0	Arrancar Replegar	Parar	Recargar
------------------	---------------------	------	---	----------------------	-------	----------

Figura 8. Aplicación desplegada.

Es mejor parar ahora la aplicación, revisar el archivo de propiedades que hay en WEB-INF/classes para que apunte a la dirección donde esté desplegados los servicios del @visorador. Una vez modificado, volver arrancar la aplicación y listo.

4.3 Demo

Para ejecutar la aplicación solo tenemos que indicar la siguiente url en nuestro navegador:

<http://servidorDeAplicaciones:Puerto/avisadorcliente/>

Tras unos instantes, saldrá la siguiente pantalla:



Figura 9. Alta/Consulta de avisos.

Si le damos a “nuevo” daremos de alta un nuevo aviso con los datos por defecto pasando a la siguiente pantalla:

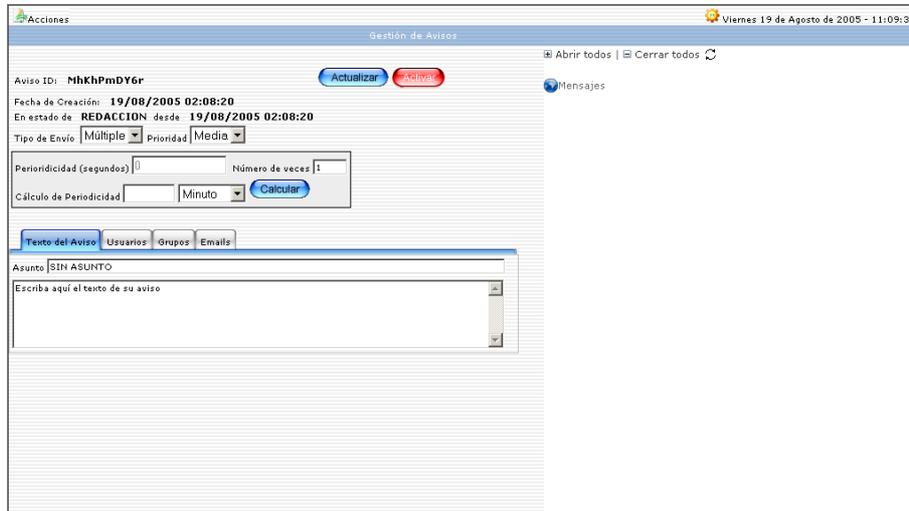


Figura 10. Alta/Modificación/Consulta de avisos

Al estar en estado de REDACCION podemos modificar los datos del mismo. Una vez que hayamos incluido todas las informaciones que queremos haciendo uso de las pestañas inferiores

En el detalle inferior hay cuatro pestañas donde podemos modificar el título o el texto del aviso e incluir destinatarios por su USUARIO, por GRUPOS o bien por direcciones de correo electrónico.

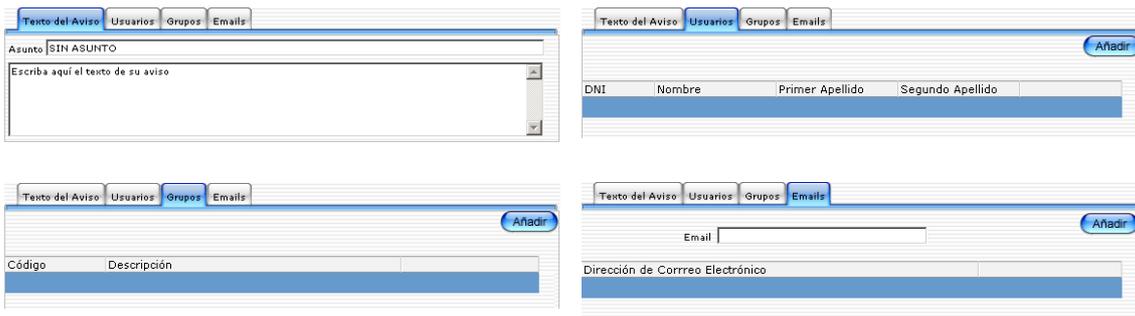


Figura 11. Título, Texto y Destinatarios del aviso.

Una vez hallamos incluido todas las informaciones a la petición pulsamos el botón Activar y esta quedará cerrada a posibles modificaciones.

Si nos apuntamos el código del aviso y pasado un tiempo lo consultamos obtendremos

en la parte de la derecha un arbol donde nos muestra el estado de entrega del aviso.



Figura 12. Aviso activo.

5 API's PLSQL

Dentro de @visorador, para la comunicación con las aplicaciones a través de la propia base de datos existe un Rol definido (AR_R_GESTION) y que la aplicación cliente deberá solicitar al administrador para poder ejecutar la API o bien trabajar con los objetos y tipos Oracle definidos.

Una vez tenga el usuario los permisos y sinónimos adecuados dado de alta podrá interactuar con la siguiente relación de paquetes.

5.1 AR_PQ_TYPE

API que contiene todos los tipos y constantes del sistema.

```
-- CONSTANTES GENERICAS
DEFAULT_RANDOM_MODO          CONSTANT VARCHAR2(1):='A';
DEFAULT_AUTO_COMMIT          CONSTANT VARCHAR2(15):='AUTO_COMMIT';
DEFAULT_RANDOM_LONG          CONSTANT NUMBER(2):=10;
DEFAULT_D_MIME                CONSTANT VARCHAR2(32):='application/octet-stream';
DEFAULT_D_ARCHIVO            CONSTANT VARCHAR2(10):='unknown';
DEFAULT_MODO                  CONSTANT VARCHAR2(32):='attachment'; -- "inline"
DEFAULT_ESPACIO              CONSTANT VARCHAR2(1):=CHR(32);
DEFAULT_RC                    CONSTANT VARCHAR2(1):=CHR(10);
DEFAULT_FB                    CONSTANT VARCHAR2(1):=CHR(13);
DEFAULT_PUNTO                 CONSTANT VARCHAR2(1):=CHR(46);
DEFAULT_FECHA                 CONSTANT VARCHAR2(25):='DD/MM/YYYY HH24:MI:SS';
DEFAULT_URL                   CONSTANT VARCHAR2(1):='/';
DEFAULT_PATH                  CONSTANT VARCHAR2(1):='\'';
DEFAULT_COMILLAS              CONSTANT VARCHAR2(1):='\"';
DEFAULT_COMILLA              CONSTANT VARCHAR2(1):=CHR(39);
DEFAULT_TAB                   CONSTANT VARCHAR2(1):=CHR(9);
DEFAULT_SEPARADOR            CONSTANT VARCHAR2(1):='%';
DEFAULT_INTERROGACION        CONSTANT VARCHAR2(1):='?';
DEFAULT_CREA_USUARIO         CONSTANT BOOLEAN:=TRUE;
DEFAULT_D_TITULO              CONSTANT VARCHAR2(20):='Desconocido';
DEFAULT_D_NOMBRE              CONSTANT VARCHAR2(20):='Desconocido';
DEFAULT_CERT_DNI              CONSTANT VARCHAR2(3):='NIF';
DEFAULT_PATH_INFO            CONSTANT VARCHAR2(10):='PATH_INFO';
DEFAULT_ORDEN                 CONSTANT NUMBER(10):=-1;
DEFAULT_PAGINA                CONSTANT NUMBER(10):=1;
DEFAULT_REGISTROS_PAGINA     CONSTANT NUMBER(10):=25;
DEFAULT_ENUMERA              CONSTANT VARCHAR2(1):='S';

-- CONSTANTES DE SISTEMA
SERVER_WEB                    CONSTANT AR_CONSTANTES.C_CONSTANTE%TYPE:=SERVER_WEB;
SERVER_IAS                    CONSTANT AR_CONSTANTES.C_CONSTANTE%TYPE:=AR_PQ_TYPE.SERVER_WEB;
DOWNLOAD_WEB                  CONSTANT AR_CONSTANTES.C_CONSTANTE%TYPE:=DOWNLOAD_WEB;
DOWNLOAD_EMAIL                CONSTANT AR_CONSTANTES.C_CONSTANTE%TYPE:=DOWNLOAD_EMAIL;

-- CONSTANTES GENERALES
CONS_AR_N_PERIODO             CONSTANT AR_CONSTANTES.C_CONSTANTE%TYPE:=AR_N_PERIODO;
CONS_AR_C_TIPO                CONSTANT AR_CONSTANTES.C_CONSTANTE%TYPE:=AR_C_TIPO;
CONS_AR_N_PRIORIDAD           CONSTANT AR_CONSTANTES.C_CONSTANTE%TYPE:=AR_N_PRIORIDAD;
CONS_TITULO                   CONSTANT AR_CONSTANTES.C_CONSTANTE%TYPE:=TITULO;
CONS_AR_D_ASUNTO              CONSTANT AR_AVISOS.D_ASUNTO%TYPE:=SIN_ASUNTO;
CONS_AR_N_VECES               CONSTANT AR_AVISOS.N_VECES%TYPE:=1;

-- CONSTANTES DE VALORES DE RETORNO EMAIL
NOTIF_LECTURA                CONSTANT AR_CONSTANTES.C_CONSTANTE%TYPE:=NOTIF_LECTURA;

-- CONSTANTES DE ESTADOS
EST_REDACCION                 CONSTANT AR_ESTADOS.C_ESTADO%TYPE:=REDACCION;
```

```

EST_DEVUELTO CONSTANT AR_ESTADOS.C_ESTADO%TYPE:='DEVUELTO';
EST_ENVIADO CONSTANT AR_ESTADOS.C_ESTADO%TYPE:='ENVIADO';
EST_LEIDO CONSTANT AR_ESTADOS.C_ESTADO%TYPE:='LEIDO';
EST_PASIVADO CONSTANT AR_ESTADOS.C_ESTADO%TYPE:='PASIVADO';
EST_RECHAZADO CONSTANT AR_ESTADOS.C_ESTADO%TYPE:='RECHAZADO';
EST_EN_ESPERA CONSTANT AR_ESTADOS.C_ESTADO%TYPE:='EN ESPERA';

-- VARIABLES ESPECIALES

VAR_EMAIL CONSTANT AR_VARIABLES.C_VARIABLE%TYPE:='EMAIL';

-- TIPOS DE TRASMISION DE MENSAJES
MEN_UNITARIO CONSTANT AR_AVISOS.C_TIPO%TYPE:='U';
MEN_MULTIPLE CONSTANT AR_AVISOS.C_TIPO%TYPE:='M';

-- EXCEPCIONES DEL SISTEMA

EXCP_AVISO_NO_REDACCION EXCEPTION;
EXCP_AVISO_NO_ACTIVACION EXCEPTION;
EXCP_CORREO_YA_PROCESADO EXCEPTION;

-- PLANTILLAS PARA MENSAJES

PLANTILLA_01 CONSTANT AR_CONSTANTES.C_CONSTANTE%TYPE:='PLANTILLA_01';
PLANTILLA_02 CONSTANT AR_CONSTANTES.C_CONSTANTE%TYPE:='PLANTILLA_02';

-- EVENTOS

EVEN_ARRANQUE CONSTANT AR_EVENTOS.C_EVENTO%TYPE:='ARRANQUE'; -- EVENTO ARRAQUE SISTEMA
EVEN_PARADA CONSTANT AR_EVENTOS.C_EVENTO%TYPE:='PARADA'; -- EVENTO PARADA SISTEMA
EVEN_ERROR CONSTANT AR_EVENTOS.C_EVENTO%TYPE:='ERROR'; -- EVENTO ERROR
EVEN_CONSULTA CONSTANT AR_EVENTOS.C_EVENTO%TYPE:='CONSULTA'; -- EVENTO CONSULTA
EVEN_ALTA CONSTANT AR_EVENTOS.C_EVENTO%TYPE:='ALTA'; -- EVENTO ALTA
EVEN_MODIFICA CONSTANT AR_EVENTOS.C_EVENTO%TYPE:='MODIFICA'; -- EVENTO MODIFICA
EVEN_BAJA CONSTANT AR_EVENTOS.C_EVENTO%TYPE:='BAJA'; -- EVENTO BAJA
EVEN_ENTRADA CONSTANT AR_EVENTOS.C_EVENTO%TYPE:='ENTRADA'; -- EVENTO ENTRADA AL SISTEMA
EVEN_SALIDA CONSTANT AR_EVENTOS.C_EVENTO%TYPE:='SALIDA'; -- EVENTO SALIDA DEL SISTEMA
EVEN_CGI CONSTANT AR_EVENTOS.C_EVENTO%TYPE:='CGI'; -- EVENTO ENTORNO CLIENTE CGI
EVEN_CORREO CONSTANT AR_EVENTOS.C_EVENTO%TYPE:='CORREO'; -- EVENTO DE GESTIÓN DE CORREO
EVEN_HTTP CONSTANT AR_EVENTOS.C_EVENTO%TYPE:='HTTP'; -- EVENTO DE GESTIÓN DE HTTP
EVEN_PROCESADO CONSTANT AR_EVENTOS.C_EVENTO%TYPE:='PROCESADO'; -- EVENTO DE PROCESADO DE
PETICION
EVEN_NOTIFICADO CONSTANT AR_EVENTOS.C_EVENTO%TYPE:='NOTIFICADO'; -- EVENTO DE NOTIFICACION

-- CONSTANTES Y OPERADORES HTTP

HTTP CONSTANT VARCHAR2(4):='HTTP'; -- PROTOCOLO DE COMUNICACION
HTTP_PORT CONSTANT NUMBER:=80; -- PUERTO DE CONEXION
HTTP_HTML_PAGE CONSTANT VARCHAR2(10):='TEXT/HTML'; -- TIPO MIME DE PÁGINAS HTML
-- SEPARADORES
HTTP_SEPA CONSTANT VARCHAR2(3):='//'; -- SEPARADORES PROTOCOLO/MAQUINA
HTTP_HEAD_SEPA CONSTANT VARCHAR2(10):=': '; -- SEPARADOR ENTRE ETIQUETAS/VALOR
HTTP_PORT_SEPA CONSTANT VARCHAR2(1):=': '; -- SEPARADOR MAQUINA/PUERTO
-- METODOS Y STATUS CODES
HTTP_MODALO CONSTANT VARCHAR2(8):='HTTP/1.1'; -- PROTOCOLO DE COMUNICACION
HTTP_GET CONSTANT VARCHAR2(3):='GET'; -- METODO GET/REQUEST
HTTP_HEAD CONSTANT VARCHAR2(4):='HEAD'; -- METODO HEAD, DEVUELVE LA CABECERA UNICAMENTE
HTTP_2XX CONSTANT VARCHAR2(2):='2%'; -- CÓDIGO: SUCCESS
HTTP_3XX CONSTANT VARCHAR2(2):='3%'; -- CÓDIGO: MOVED
HTTP_4XX CONSTANT VARCHAR2(2):='4%'; -- CÓDIGO: ERROR
HTTP_5XX CONSTANT VARCHAR2(2):='5%'; -- CÓDIGO: FATAL ERROR
-- VALORES PARA EL "GET"
HTTP_LANG_ACCEPT CONSTANT VARCHAR2(20):='ACCEPT-LANGUAGE';
HTTP_MIME_ACCEPT CONSTANT VARCHAR2(10):='ACCEPT';
HTTP_CHAR_ACCEPT CONSTANT VARCHAR2(20):='ACCEPT-CHARSET';
HTTP_USER_AGENT CONSTANT VARCHAR2(10):='USER-AGENT';
HTTP_HOST CONSTANT VARCHAR2(10):='HOST';
HTTP_CONNECTION CONSTANT VARCHAR2(10):='CONNECTION';
-- VALORES DE RETORNO "GET"

HTTP_MIME CONSTANT VARCHAR2(20):='CONTENT-TYPE'; -- ETIQUETA DEL TIPO MIME
HTTP_MOVE CONSTANT VARCHAR2(10):='LOCATION'; -- ETIQUETA DE NUEVA LOCALIZACION
HTTP_DISP CONSTANT VARCHAR2(30):='CONTENT-DISPOSITION'; -- ETIQUETA PARA INDICAR EL MODO DE APERTURA
Y EL NOMBRE DEL ARCHIVO
HTTP_FILE CONSTANT VARCHAR2(10):='FILENAME'; -- ETIQUETA PARA INDICAR EL NOMBRE DEL ARCHIVO
HTTP_TRANS CONSTANT VARCHAR2(30):='TRANSFER-ENCODING'; -- TIPO DE TRANSFERENCIA
HTTP_CHUNK CONSTANT VARCHAR2(10):='CHUNKED'; -- MODO A CACHITOS

-- ETIQUETAS Y CAMPOS CLAVE DE HTML

HTML_ERROR CONSTANT VARCHAR2(20):='HTML_ERROR';
HTML_MODALO CONSTANT VARCHAR2(20):='HTML_MODALO';

```

```

HTML_MODO_ABS CONSTANT VARCHAR2(1):='A';
HTML_MODO_REL CONSTANT VARCHAR2(1):='R';
HTML_MODO_DIR CONSTANT VARCHAR2(1):='D';
HTML_BASE CONSTANT VARCHAR2(10):='<BASE ';
HTML_HREF CONSTANT VARCHAR2(10):='HREF=';
HTML_IMG CONSTANT VARCHAR2(10):='<IMG ';
HTML_LINK CONSTANT VARCHAR2(10):='<LINK ';
HTML_FRAME CONSTANT VARCHAR2(10):='<FRAME ';
HTML_IFRAME CONSTANT VARCHAR2(10):='<IFRAME ';
HTML_SRC CONSTANT VARCHAR2(10):='SRC=';
HTML_BODY CONSTANT VARCHAR2(10):='<BODY ';
HTML_BODY_FIN CONSTANT VARCHAR2(10):='</BODY';
HTML_BACK CONSTANT VARCHAR2(20):='BACKGROUND=';
HTML_TABLE CONSTANT VARCHAR2(10):='<TABLE ';
HTML_ROW CONSTANT VARCHAR2(5):='<TR ';
HTML_COLUMN CONSTANT VARCHAR2(5):='<TD ';
HTML_FORM CONSTANT VARCHAR2(10):='<FORM ';
HTML_FORM_FIN CONSTANT VARCHAR2(10):='</FORM';
HTML_INPUT CONSTANT VARCHAR2(10):='<INPUT ';
HTML_NAME CONSTANT VARCHAR2(10):='NAME=';
HTML_FIN CONSTANT VARCHAR2(1):='>';

-- CORREO

CONS_SMTP CONSTANT AR_CONSTANTES.C_CONSTANTE%TYPE:='EMAIL_SMTP';
CONS_POP3 CONSTANT AR_CONSTANTES.C_CONSTANTE%TYPE:='EMAIL_POP3';
CONS_HELO CONSTANT AR_CONSTANTES.C_CONSTANTE%TYPE:='EMAIL_HELO';
CONS_USER CONSTANT AR_CONSTANTES.C_CONSTANTE%TYPE:='EMAIL_USER';
CONS_PASS CONSTANT AR_CONSTANTES.C_CONSTANTE%TYPE:='EMAIL_PASS';
CONS_EMAIL CONSTANT AR_CONSTANTES.C_CONSTANTE%TYPE:='EMAIL';
CONS_NOMBRE CONSTANT AR_CONSTANTES.C_CONSTANTE%TYPE:='EMAIL_NOMBRE';

-- CONSTANTES Y OPERADORES POP3

POP3_PORT CONSTANT INTEGER:=110; -- PUERTO PROTOCOLO POP3
POP3_RESU CONSTANT INTEGER:=50; -- NUMERO DE LINEAS PARA UN RESUMEN
POP3_OK CONSTANT VARCHAR2(3):='+OK'; -- CORRECTO
POP3_ER CONSTANT VARCHAR2(4):='ERR'; -- ERROR
POP3_USER CONSTANT VARCHAR2(4):='USER'; -- USUARIO
POP3_PASS CONSTANT VARCHAR2(4):='PASS'; -- CLAVE
POP3_STAT CONSTANT VARCHAR2(4):='STAT'; -- ESTATUS DEL BUZON
POP3_QUIT CONSTANT VARCHAR2(4):='QUIT'; -- CERRAR LA CONEXIÓN POP
POP3_LIST CONSTANT VARCHAR2(4):='LIST'; -- LISTA DE CORREOS
POP3_RETR CONSTANT VARCHAR2(4):='RETR'; -- RECUPERA EL MENSAJE MARCADO
POP3_DELE CONSTANT VARCHAR2(4):='DELE'; -- BORRA EL MENSAJE MARCADO
POP3_NOOP CONSTANT VARCHAR2(4):='NOOP'; -- PARA COMPROBAR SI EL SERVIDOR ESTA ON-LINE, NO HACE NADA.
-- EL COMMANDO TOP NO ES PROPIO DEL ESTANDAR, PERO CASI LA TOTALIDAD DE LOS SERVIDORES LO ADMITEN
COMO VALIDO
POP3_TOP CONSTANT VARCHAR2(4):='TOP'; -- RECUPERA LA CABECERA Y LAS 10 PRIMERAS LINEAS DE CORREO
MARCADAO

-- CONSTANTES Y OPERADORS CORREO MIME V1.0
MIME_FROM CONSTANT VARCHAR2(30):='From';
MIME_TO CONSTANT VARCHAR2(10):='To';
MIME_CC CONSTANT VARCHAR2(10):='Cc';
MIME_CCO CONSTANT VARCHAR2(10):='Cco';
MIME_SUBJECT CONSTANT VARCHAR2(20):='Subject';
MIME_MAILER CONSTANT VARCHAR2(30):='X-Mailer';
MIME_MAILER_GTEL CONSTANT VARCHAR2(50):='Servidor de correo Oracle - GuadalTEL S.A. v1.0.0';
MIME_PRIORIDAD CONSTANT VARCHAR2(30):='X-Priority';
-- Alta Normal Baja
-- 1 2 3 4 5
MIME_PRIORIDAD_DEFAULT CONSTANT NUMBER(1):=3;
MIME CONSTANT VARCHAR2(30):='MIME-Version';
MIME_VERSION CONSTANT VARCHAR2(30):='1.0';
MIME_TYPE CONSTANT VARCHAR2(30):='Content-Type';
MIME_MULTIPART_MIXED CONSTANT VARCHAR2(30):='multipart/mixed';
MIME_MULTIPART_ALTERNATIVE CONSTANT VARCHAR2(30):='multipart/alternative';
MIME_MULTIPART_DIALOG CONSTANT VARCHAR2(50):='This is a multi-part message in MIME format.';
MIME_TYPE_TEXTPLAIN CONSTANT VARCHAR2(20):='text/plain';
MIME_CHARSET CONSTANT VARCHAR2(30):='charset';
MIME_ISO CONSTANT VARCHAR2(30):='iso-8859-1';
MIME_TYPE_MESSAGE CONSTANT VARCHAR2(30):='message/rfc822';
MIME_NAME CONSTANT VARCHAR2(30):='name'; -- EL NOMBRE ENTRE COMILLAS DOBLES
MIME_TRANSFER_ENCODING CONSTANT VARCHAR2(50):='Content-Transfer-Encoding';
MIME_ENCODING_BASE64 CONSTANT VARCHAR2(30):='base64';
MIME_ENCODING_QUOTED CONSTANT VARCHAR2(30):='quoted-printable';
MIME_ENCODING_8BIT CONSTANT VARCHAR2(30):='8bit';
MIME_ENCODING_7BIT CONSTANT VARCHAR2(30):='7bit';
MIME_CONTENT_DISPOSITION CONSTANT VARCHAR2(30):='Content-Disposition';
MIME_DEFAULT_DISPOSITION CONSTANT VARCHAR2(30):='attachment'; -- SIEMPRE SERÁ ATTACHMENT
MIME_FILENAME CONSTANT VARCHAR2(30):='filename'; -- EL NOMBRE ENTRE COMILLAS DOBLES

```

```

MIME_BOUNDARY          CONSTANT VARCHAR2(30):='boundary';
MIME_INI_BOUNDARY      CONSTANT VARCHAR2(50) := '-----_NextPart_000_0001_'; -- HAY QUE AÑADIRLE 2
RANDOM DE 8 SEPARADOS POR UN PUNTO
MIME_FIN_BOUNDARY      CONSTANT VARCHAR2(2) := '-';
MIME_RANDOM_LEN        CONSTANT NUMBER(1):=8;
MIME_NOTIFICAR_LECTURA CONSTANT VARCHAR2(30):='Disposition-Notification-To';

-- TIPOS PARA LA CONSTRUCCION DINAMICA DE SELECT¿S

TYPE TYPE_COLUMNA IS RECORD
(
    D_NOMBRE VARCHAR2(60),
    D_TIPO   VARCHAR2(30),
    N_TAMANIO NUMBER
);

TYPE ARRAY_COLUMNA IS TABLE OF TYPE_COLUMNA
INDEX BY BINARY_INTEGER;

-- TIPOS ESPECIALES PARA PROCESAMIENTO DE DOCUMENTOS, CORREO Y WEB

TYPE REC_LISTA_D_EMAIL IS RECORD
(
    D_EMAIL AR_USUARIOS_EMAIL.D_EMAIL%TYPE
);

TYPE ARRAY_LISTA_D_EMAIL IS TABLE OF REC_LISTA_D_EMAIL
INDEX BY BINARY_INTEGER;

TYPE REC_LISTA_CORREO IS RECORD
(
    CABECERA VARCHAR2(4000), -- CABECERA DEL CORREO
    RESUMEN  VARCHAR2(4000), -- 10 PRIMERAS LINEAS DEL CORREO
    TAMANIO  NUMBER        -- TAMAÑO COMPLETO DEL CORREO
);

TYPE ARRAY_LISTA_CORREO IS TABLE OF REC_LISTA_CORREO -- ARRAY D
INDEX BY BINARY_INTEGER;

-- ARCHIVO ADJUNTO

TYPE REC_ADJUNTOS IS RECORD
(
    D_ARCHIVO VARCHAR2(255),
    D_MIME    VARCHAR2(50),
    B_ARCHIVO BLOB
);

TYPE ARRAY_ADJUNTOS IS TABLE OF REC_ADJUNTOS -- ARRAY DE ARCHIVOS ADJUNTOS
INDEX BY BINARY_INTEGER;

TYPE REC_URL IS RECORD
(
    ORIGINAL VARCHAR2(2000),
    DESCARGA VARCHAR2(2000),
    C_HASH   VARCHAR2(10)
);

TYPE ARRAY_URL IS TABLE OF REC_URL
INDEX BY BINARY_INTEGER;

-- TIPOS GENÉRICOS

TAMANIO_T_OBSERVACIONES CONSTANT INTEGER:=1000; -- TAMAÑO DE LOS CAMPOS T_OBSERVACIONES
TAMANIO_ARRAY_TEXTO     CONSTANT INTEGER:=2000; -- TAMAÑO DEL BUFFER DEL ARRAY
TAMANIO_ARRAY_RAW       CONSTANT INTEGER:=2000; -- TAMAÑO DEL BUFFER DEL ARRAY

TYPE ARRAY_TEXTO IS TABLE OF VARCHAR2(2000) -- ARRAY DE VARCHAR2
INDEX BY BINARY_INTEGER;

TYPE ARRAY_RAW IS TABLE OF RAW(2000)
INDEX BY BINARY_INTEGER;

TYPE ARRAY_BLOB IS TABLE OF BLOB
INDEX BY BINARY_INTEGER;

/*-----*/
/* FU_CONSTANTE */
/*-----*/
FUNCTION FU_CONSTANTE(ECONSTANTE IN VARCHAR2) RETURN VARCHAR2;
-- DEVUELVE EL VALOR DE UNA CONSTANTE

```

```
-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- ECONSTANTE - VARCHAR2 - NOMBRE DE LA CONSTANTE

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

-- VALORES DEVUELTOS
-- VARCHAR2 - VALOR DE LA CONSTANTE

-- ERRORS:
-- NO_DATA_FOUND - LA CONSTANTE REFERIDA NO EXISTE

/*-----*/
/* PR_CONSTANTE */
/*-----*/
PROCEDURE PR_CONSTANTE(ECONSTANTE IN VARCHAR2,CONSTANTE IN OUT COL_AR_CONSTANTES);
-- DEVUELVE LOS VALORES DE UNA CONSTANTE

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- ECONSTANTE - VARCHAR2 - NOMBRE DE LA CONSTANTE
-- CONSTANTE - COL_AR_CONSTANTES

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

-- VALORES DEVUELTOS
-- CONSTANTE - COL_AR_CONSTANTES - CON TODOS LOS VALORES DE LA CONSTANTE

-- ERRORS:
-- NO_DATA_FOUND - LA CONSTANTE REFERIDA NO EXISTE
```

5.2 AR_PQ_AVISADOR

API de grabación/consulta de avisos.

```
/*-----*/
/* FU_AVISO_NUEVO */
/*-----*/
FUNCTION FU_AVISO_NUEVO(T_AVISO IN VARCHAR2,
                        N_PERIODICIDAD IN NUMBER DEFAULT NULL,
                        C_TIPO IN VARCHAR2 DEFAULT NULL,
                        N_PRIORIDAD IN NUMBER DEFAULT NULL,
                        D_ASUNTO IN VARCHAR2 DEFAULT NULL,
                        N_VECES IN NUMBER DEFAULT NULL) RETURN
VARCHAR2;
-- DEVUELVE EL CÓDIGO HASH DEL AVISO QUE SE ACABA DE CREAR

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- T_AVISO - VARCHAR2 - TEXTO DEL AVISO A CREAR

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- N_PERIODICIDAD - NUMBER - PERIODICIDAD DE ENVÍO DEL AVISO
-- C_TIPO - VARCHAR2 - TIPO DE ENVÍO DEL AVISO (MULTIPLE O INDIVIDUAL)
-- N_PRIORIDAD - NUMBER - PRIORIDAD DEL ENVÍO
-- D_ASUNTO - VARCHAR2 - ASUNTO DEL AVISO
-- N_VECES - NUMBER - VECES EN LA PERIODICIDAD

-- VALORES DEVUELTOS
-- VARCHAR2 - CÓDIGO HASH DEL AVISO QUE SE ACABA DE CREAR

/*-----*/
/* FU_AVISO_NUEVO */
/*-----*/
FUNCTION FU_AVISO_NUEVO(AVISO OBJ_AR_AVISOS) RETURN VARCHAR2;
-- DEVUELVE EL CÓDIGO HASH DEL AVISO QUE SE ACABA DE CREAR

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- AVISO - OBJ_AR_AVISOS - OBJETO CON LOS VALORES PARA CREAR EL AVISO

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

-- VALORES DEVUELTOS
-- VARCHAR2 - CÓDIGO HASH DEL AVISO QUE SE ACABA DE CREAR

/*-----*/
/* PR_AVISO_MODIFICA */
/*-----*/
PROCEDURE PR_AVISO_MODIFICA(C_HASH IN VARCHAR2,
```

```

NULL,
NULL,
NULL,
FALSE);
-- PROCEDIMIENTO QUE MODIFICA EL AVISO

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- C_HASH - VARCHAR2 - CÓDIGO HASH DEL AVISO A MODIFICAR
-- T_AVISO - VARCHAR2 - TEXTO DEL AVISO A CREAR

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- N_PERIODICIDAD - NUMBER - PERIODICIDAD DE ENVÍO DEL AVISO
-- C_TIPO - VARCHAR2 - TIPO DE ENVÍO DEL AVISO (MULTIPLE O INDIVIDUAL)
-- N_PRIORIDAD - NUMBER - PRIORIDAD DEL ENVÍO
-- D_ASUNTO - VARCHAR2 - ASUNTO DEL AVISO
-- N_VECES - NUMBER - VECES EN LA PERIODICIDAD
-- PERMITE_NULOS - BOOLEAN - DICE SI SE PERMITE NULOS O NO

/*-----*/
/* PR_AVISO_MODIFICA */
/*-----*/
PROCEDURE PR_AVISO_MODIFICA(AVISO OBJ_AR_AVISOS,PERMITE_NULOS IN BOOLEAN DEFAULT FALSE);
-- PROCEDIMIENTO QUE MODIFICA EL AVISO

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- AVISO - OBJ_AR_AVISOS - OBJETO CON LOS VALORES PARA MODIFICAR EL AVISO
-- PERMITE_NULOS - BOOLEAN - DICE SI SE PERMITE NULOS O NO

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

/*-----*/
/* PR_AVISO_ANADE_USUARIO */
/*-----*/
PROCEDURE PR_AVISO_ANADE_USUARIO(USUARIO IN VARCHAR2,AVISO IN VARCHAR2);
-- PROCEDIMIENTO QUE AÑADE UN USUARIO COMO DESTINATARIO DE UN AVISO

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- USUARIO - VARCHAR2 - USUARIO QUE SE VA A AÑADIR A LA LISTA DE
DESTINATARIOS
-- AVISO - VARCHAR2 - AVISO AL QUE SE AÑADE EL USUARIO

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

/*-----*/
/* PR_AVISO_ANADE_GRUPO */
/*-----*/
PROCEDURE PR_AVISO_ANADE_GRUPO(X_GRUPO IN NUMBER, C_SISTEMA IN VARCHAR2,AVISO IN VARCHAR2);
-- PROCEDIMIENTO QUE AÑADE UN GRUPO COMO DESTINATARIO DE UN AVISO

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- X_GRUPO - NUMBER - CLAVE DEL GRUPO QUE SE VA A AÑADIR A LA LISTA DE
DESTINATARIOS
-- C_SISTEMA - VARCHAR2 - 2ª PARTE DE LA CLAVE DEL GRUPO QUE SE VA A AÑADIR
-- AVISO - VARCHAR2 - AVISO AL QUE SE AÑADE EL GRUPO

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

/*-----*/
/* PR_AVISO_ANADE_GRUPO */
/*-----*/
PROCEDURE PR_AVISO_ANADE_GRUPO(C_GRUPO IN VARCHAR2,AVISO IN VARCHAR2);
-- PROCEDIMIENTO QUE AÑADE UN GRUPO COMO DESTINATARIO DE UN AVISO

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- C_GRUPO - VARCHAR2 - CÓDIGO DEL GRUPO QUE SE VA A AÑADIR A LA LISTA DE
DESTINATARIOS
-- AVISO - VARCHAR2 - AVISO AL QUE SE AÑADE EL GRUPO

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

/*-----*/

```

```

/* PR_Aviso_Añade_Email */
/*-----*/
PROCEDURE PR_Aviso_Añade_Email(EMAIL IN VARCHAR2,AVISO IN VARCHAR2);
-- PROCEDIMIENTO QUE AÑADE UN EMAIL COMO DESTINATARIO DE UN AVISO

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- EMAIL - VARCHAR2 - EMAIL QUE SE VA A AÑADIR A LA LISTA DE DESTINATARIOS
-- AVISO - VARCHAR2 - AVISO AL QUE SE AÑADE EL EMAIL

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

/*-----*/
/* PR_Aviso_Añade_Movil */
/*-----*/
PROCEDURE PR_Aviso_Añade_Movil(MOVIL IN VARCHAR2,AVISO IN VARCHAR2);
-- PROCEDIMIENTO QUE AÑADE UN MOVIL COMO DESTINATARIO DE UN AVISO

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- MOVIL - VARCHAR2 - MOVIL QUE SE VA A AÑADIR A LA LISTA DE DESTINATARIOS
-- AVISO - VARCHAR2 - AVISO AL QUE SE AÑADE EL MOVIL

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

/*-----*/
/* PR_Aviso_Elimina_Usuario */
/*-----*/
PROCEDURE PR_Aviso_Elimina_Usuario(USUARIO IN VARCHAR2,AVISO IN VARCHAR2);
-- PROCEDIMIENTO QUE ELIMINA UN USUARIO COMO DESTINATARIO DE UN AVISO

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- USUARIO - VARCHAR2 - USUARIO QUE SE VA A ELIMINAR DE LA LISTA DE
DESTINATARIOS
-- AVISO - VARCHAR2 - AVISO DEL QUE SE ELIMINA EL USUARIO

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

/*-----*/
/* PR_Aviso_Elimina_Grupo */
/*-----*/
PROCEDURE PR_Aviso_Elimina_Grupo(X_GRUPO IN NUMBER, C_SISTEMA IN VARCHAR2,AVISO IN
VARCHAR2);
-- PROCEDIMIENTO QUE ELIMINA UN GRUPO COMO DESTINATARIO DE UN AVISO

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- X_GRUPO - NUMBER - CLAVE DEL GRUPO QUE SE VA A ELIMINAR DE LA LISTA DE
DESTINATARIOS
-- C_SISTEMA - VARCHAR2 - 2ª PARTE DE LA CLAVE DEL GRUPO QUE SE VA A ELIMINAR
-- AVISO - VARCHAR2 - AVISO DEL QUE SE ELIMINA EL GRUPO

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

/*-----*/
/* PR_Aviso_Elimina_Grupo */
/*-----*/
PROCEDURE PR_Aviso_Elimina_Grupo(C_GRUPO IN VARCHAR2,AVISO IN VARCHAR2);
-- PROCEDIMIENTO QUE ELIMINA UN GRUPO COMO DESTINATARIO DE UN AVISO

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- C_GRUPO - VARCHAR2 - CÓDIGO DEL GRUPO QUE SE VA A ELIMINAR DE LA LISTA DE
DESTINATARIOS
-- AVISO - VARCHAR2 - AVISO DEL QUE SE ELIMINA EL GRUPO

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

/*-----*/
/* PR_Aviso_Elimina_Email */
/*-----*/
PROCEDURE PR_Aviso_Elimina_Email(EMAIL IN VARCHAR2,AVISO IN VARCHAR2);
-- PROCEDIMIENTO QUE ELIMINA UN EMAIL COMO DESTINATARIO DE UN AVISO

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- EMAIL - VARCHAR2 - EMAIL QUE SE VA A ELIMINAR DE LA LISTA DE
DESTINATARIOS
-- AVISO - VARCHAR2 - AVISO DEL QUE SE ELIMINA EL EMAIL

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

```

```
/*-----*/
/* PR_Aviso_ELIMINA_MOVIL */
/*-----*/
PROCEDURE PR_Aviso_ELIMINA_MOVIL(MOVI IN VARCHAR2,AVISO IN VARCHAR2);
-- PROCEDIMIENTO QUE ELIMINA UN MOVIL COMO DESTINATARIO DE UN AVISO

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- MOVIL - VARCHAR2 - MOVIL QUE SE VA A ELIMINAR DE LA LISTA DE
DESTINATARIOS
-- AVISO - VARCHAR2 - AVISO DEL QUE SE ELIMINA EL MOVIL

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

/*-----*/
/* PR_Aviso_CONSULTA_USUARIOS */
/*-----*/
PROCEDURE PR_Aviso_CONSULTA_USUARIOS(USUARIOS_DESTINATARIO IN OUT
COL_AR_USUARIOS_DESTINATARIO,AVISO IN VARCHAR2);
-- PROCEDIMIENTO QUE DEVUELVE LOS USUARIOS QUE ESTAN ASOCIADOS A UN AVISO

-- PARAMETROS DE ENTRADA/SALIDA OBLIGATORIOS
-- USUARIOS_DESTINATARIO - COL_USUARIOS_DESTINATARIO - COLECCIÓN DONDE SE VAN DEVOLVER
LOS USUARIOS

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- AVISO - VARCHAR2 - AVISO DEL CUAL SE QUIEREN SABER SUS USUARIOS ASOCIADOS

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

/*-----*/
/* PR_Aviso_CONSULTA_GRUPOS */
/*-----*/
PROCEDURE PR_Aviso_CONSULTA_GRUPOS(GRUPOS_DESTINATARIO IN OUT COL_AR_GRUPOS_DESTINATARIO,AVISO
IN VARCHAR2);
-- PROCEDIMIENTO QUE DEVUELVE LOS GRUPOS QUE ESTAN ASOCIADOS A UN AVISO

-- PARAMETROS DE ENTRADA/SALIDA OBLIGATORIOS
-- GRUPOS_DESTINATARIO - COL_GRUPOS_DESTINATARIO - COLECCIÓN DONDE SE VAN DEVOLVER LOS
GRUPOS

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- AVISO - VARCHAR2 - AVISO DEL CUAL SE QUIEREN SABER SUS GRUPOS ASOCIADOS

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

/*-----*/
/* PR_Aviso_CONSULTA_EMAIL */
/*-----*/
PROCEDURE PR_Aviso_CONSULTA_EMAIL(EMAILS_DESTINATARIO IN OUT COL_AR_EMAIL_DESTINATARIO,AVISO IN
VARCHAR2);
-- PROCEDIMIENTO QUE DEVUELVE LOS EMAILS QUE ESTAN ASOCIADOS A UN AVISO

-- PARAMETROS DE ENTRADA/SALIDA OBLIGATORIOS
-- EMAILS_DESTINATARIO - COL_EMAIL_DESTINATARIO - COLECCIÓN DONDE SE VAN DEVOLVER LOS EMAILS

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- AVISO - VARCHAR2 - AVISO DEL CUAL SE QUIEREN SABER SUS EMAILS ASOCIADOS

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

/*-----*/
/* PR_Aviso_CONSULTA_MOVIL */
/*-----*/
PROCEDURE PR_Aviso_CONSULTA_MOVIL(MOVILS_DESTINATARIO IN OUT COL_AR_MOVIL_DESTINATARIO,AVISO IN
VARCHAR2);
-- PROCEDIMIENTO QUE DEVUELVE LOS MOVILES QUE ESTAN ASOCIADOS A UN AVISO

-- PARAMETROS DE ENTRADA/SALIDA OBLIGATORIOS
-- MOVILS_DESTINATARIO - COL_MOVIL_DESTINATARIO - COLECCIÓN DONDE SE VAN DEVOLVER LOS MOVILES

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- AVISO - VARCHAR2 - AVISO DEL CUAL SE QUIEREN SABER SUS MOVILES ASOCIADOS

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

/*-----*/
```

```
/* FU_AVISO_ESTADO_CONSULTA */
/*-----*/
FUNCTION FU_AVISO_ESTADO_CONSULTA(AVISO IN VARCHAR2) RETURN VARCHAR2;
-- FUNCIÓN QUE DEVUELVE EL ESTADO EN EL QUE SE ENCUENTRA UN AVISO

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- AVISO - VARCHAR2 - AVISO DEL QUE SE QUIERE CONOCER SU ESTADO

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

-- VALORES DEVUELTOS
-- VARCHAR2 - ESTADO DEL AVISO

/*-----*/
/* PR_AVISO_ESTADO_ACTIVACION */
/*-----*/
PROCEDURE PR_AVISO_ESTADO_ACTIVACION(AVISO IN VARCHAR2);
-- PROCEDIMIENTO QUE ACTIVA UN AVISO (PASA EL ESTADO DE REDACCIÓN A EN ESPERA)

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- AVISO - VARCHAR2 - AVISO QUE SE QUIERE ACTIVAR

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

-- ERRORS:
-- EXCP_AVISO_NO_REDACCION: EL ESTADO DEL AVISO ES DISTINTO A REDACCION
-- (NO SE ACTIVARÁ)
-- EXCP_AVISO_NO_ACTIVACION: EL AVISO NO SE PUEDE ACTIVAR POR:
-- - NO HAY TEXTO
-- - NO HAY NINGUN DESTINATARIO (USUARIO O
GRUPO)

/*-----*/
/* PR_AVISO_ESTADO_REDACCION */
/*-----*/
PROCEDURE PR_AVISO_ESTADO_REDACCION(AVISO IN VARCHAR2);
-- PROCEDIMIENTO QUE PASA AL ESTADO DE REDACCIÓN

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- AVISO - VARCHAR2 - AVISO QUE SE QUIERE CAMBIAR DE ESTADO

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

/*-----*/
/* PR_AVISO_ESTADO_ENVIADO */
/*-----*/
PROCEDURE PR_AVISO_ESTADO_ENVIADO(AVISO IN VARCHAR2);
-- PROCEDIMIENTO QUE PASA AL ESTADO DE ENVIADO

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- AVISO - VARCHAR2 - AVISO QUE SE QUIERE CAMBIAR DE ESTADO

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

/*-----*/
/* PR_AVISO_CONSULTA */
/*-----*/
PROCEDURE PR_AVISO_CONSULTA(AVISOS IN OUT COL_AR_AVISOS,AVISO IN VARCHAR2);
-- PROCEDIMIENTO QUE DEVUELVE UNA COLECCION CON EL AVISO CUYO C_HASH ES AVISO

-- PARAMETROS DE ENTRADA/SALIDA OBLIGATORIOS
-- AVISOS - COL_AR_AVISOS - COLECCION DONDE SE DEVUELVE EL AVISO

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- AVISO - VARCHAR2 - AVISO QUE SE QUIERE CONSULTAR

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

/*-----*/
/* PR_USUARIO_CONSULTA */
/*-----*/
PROCEDURE PR_USUARIO_CONSULTA(USUARIO IN VARCHAR2,DNI IN OUT VARCHAR2,NOMBRE IN OUT
VARCHAR2,
APEL1 IN OUT VARCHAR2,APEL2 IN OUT VARCHAR2);
-- PROCEDIMIENTO QUE DEVUELVE TODOS LOS CAMPOS DE ESE USUARIO
```

```
-- PARAMETROS DE ENTRADA/SALIDA OBLIGATORIOS
--     DNI      - VARCHAR2 - DNI DEL USUARIO
--     NOMBRE   - VARCHAR2 - NOMBRE
--     APELL1   - VARCHAR2   - PRIMER APELLIDO
--     APELL2   - VARCHAR2   - SEGUNDO APELLIDO

-- PARAMETROS DE ENTRADA OBLIGATORIOS
--     USUARIO  - VARCHAR2   - USUARIO QUE SE QUIERE CONSULTAR

-- VALORES POR DEFECTO O VALORES OPCIONALES
--     NINGUNO

/*-----*/
/* PR_GRUPO_CONSULTA                               */
/*-----*/
PROCEDURE PR_GRUPO_CONSULTA(GRUPO IN NUMBER,SISTEMA IN VARCHAR2,C_GRUPO IN OUT
VARCHAR2,D_GRUPO IN OUT VARCHAR2);
-- PROCEDIMIENTO QUE DEVUELVE TODOS LOS CAMPOS DE ESE GRUPO

-- PARAMETROS DE ENTRADA/SALIDA OBLIGATORIOS
--     C_GRUPO  - VARCHAR2 - CÓDIGO DEL GRUPO
--     D_GRUPO  - VARCHAR2 - DESCRIPCIÓN DEL GRUPO

-- PARAMETROS DE ENTRADA OBLIGATORIOS
--     GRUPO    - VARCHAR2   - GRUPO QUE SE QUIERE CONSULTAR

-- VALORES POR DEFECTO O VALORES OPCIONALES
--     NINGUNO

/*-----*/
/* PR_USUARIOS_CONSULTA                             */
/*-----*/
PROCEDURE PR_USUARIOS_CONSULTA(USUARIOS IN OUT COL_AR_USUARIOS, QUERY IN VARCHAR2);
-- PROCEDIMIENTO QUE DEVUELVE TODOS LOS USUARIOS

-- PARAMETROS DE ENTRADA/SALIDA OBLIGATORIOS
--     USUARIOS - COL_AR_USUARIOS - DONDE SE VA A DEVOLVER TODOS LOS USUARIOS

-- VALORES POR DEFECTO O VALORES OPCIONALES
--     QUERY   - VARCHAR2           - CONSULTA

/*-----*/
/* PR_GRUPOS_CONSULTA                               */
/*-----*/
PROCEDURE PR_GRUPOS_CONSULTA(GRUPOS IN OUT COL_AR_GRUPOS, QUERY IN VARCHAR2);
-- PROCEDIMIENTO QUE DEVUELVE TODOS LOS GRUPOS

-- PARAMETROS DE ENTRADA/SALIDA OBLIGATORIOS
--     GRUPOS  - COL_AR_GRUPOS - DONDE SE VA A DEVOLVER TODOS LOS GRUPOS

-- VALORES POR DEFECTO O VALORES OPCIONALES
--     QUERY  - VARCHAR2           - CONSULTA

/*-----*/
/* PR_MENSAJES_CONSULTA                             */
/*-----*/
PROCEDURE PR_MENSAJES_CONSULTA(MENSAJES IN OUT COL_AR_MENSAJES, AVISO IN VARCHAR2);
-- PROCEDIMIENTO QUE DEVUELVE TODOS LOS MENSAJES DE ESE AVISO

-- PARAMETROS DE ENTRADA/SALIDA OBLIGATORIOS
--     MENSAJES - COL_AR_MENSAJES - DONDE SE VA A DEVOLVER TODOS LOS MENSAJES

-- VALORES POR DEFECTO O VALORES OPCIONALES
--     AVISO    - VARCHAR2           - AVISO DEL QUE QUEREMOS RECUPERAR LOS MENSAJES

/*-----*/
/* PR_MENSAJES_CONSULTA                             */
/*-----*/
PROCEDURE PR_MENSAJES_CONSULTA(MENSAJES IN OUT COL_AR_MENSAJES, MENSAJE IN NUMBER);
-- PROCEDIMIENTO QUE DEVUELVE LOS CAMPOS DE ESE MENSAJE

-- PARAMETROS DE ENTRADA/SALIDA OBLIGATORIOS
--     MENSAJES - COL_AR_MENSAJES - DONDE SE VA A DEVOLVER EL MENSAJE QUE QUEREMOS

-- VALORES POR DEFECTO O VALORES OPCIONALES
--     MENSAJE  - NUMBER             - MENSAJE DEL QUE QUEREMOS RECUPERAR SUS
CAMPOS

/*-----*/
/* PR_ENVIOS_EMAIL_CONSULTA                         */
/*-----*/
PROCEDURE PR_ENVIOS_EMAIL_CONSULTA(ENVIOS_EMAIL IN OUT COL_AR_ENVIOS_EMAIL, MENSAJE IN
```

```

VARCHAR2);
-- PROCEDIMIENTO QUE DEVUELVE TODOS LOS ENVIOS_EMAIL DE ESE MENSAJE

-- PARAMETROS DE ENTRADA/SALIDA OBLIGATORIOS
-- ENVIOS_EMAIL - COL_AR_ENVIOS_EMAIL - DONDE SE VA A DEVOLVER TODOS LOS ENVIOS_EMAIL

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- MENSAJE - VARCHAR2 - MENSAJE DEL QUE QUEREMOS OBTENER LOS ENVIOS

/*-----*/
/* PR_ENVIOS_MOVIL_CONSULTA */
/*-----*/
PROCEDURE PR_ENVIOS_MOVIL_CONSULTA(ENVIOS_MOVIL IN OUT COL_AR_ENVIOS_MOVIL, MENSAJE IN
VARCHAR2);
-- PROCEDIMIENTO QUE DEVUELVE TODOS LOS ENVIOS_MOVIL DE ESE MENSAJE

-- PARAMETROS DE ENTRADA/SALIDA OBLIGATORIOS
-- ENVIOS_MOVIL - COL_AR_ENVIOS_MOVIL - DONDE SE VA A DEVOLVER TODOS LOS ENVIOS_MOVIL

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- MENSAJE - VARCHAR2 - MENSAJE DEL QUE QUEREMOS OBTENER LOS ENVIOS

/*-----*/
/* FU_ENVIOS_ESTADO_CONSULTA */
/*-----*/
FUNCTION FU_ENVIOS_ESTADO_CONSULTA(EMAIL IN VARCHAR2, MENSAJE IN VARCHAR2) RETURN VARCHAR2;
-- FUNCIÓN QUE DEVUELVE EL ESTADO EN EL QUE SE ENCUENTRA UN ENVIO

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- EMAIL - VARCHAR2 - EMAIL DEL QUE SE QUIERE SABER SU ESTADO DE ENVIO
-- MENSAJE - VARCHAR2 - MENSAJE AL QUE PERTECENECE ESE ENVÍO

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

-- VALORES DEVUELTOS
-- VARCHAR2 - ESTADO DEL ENVIO

/*-----*/
/* FU_ENVIOS_MOVIL_ESTADO_CONSULTA */
/*-----*/
FUNCTION FU_ENVIOS_MVL_ESTADO_CONSULTA(MOVIL IN VARCHAR2, MENSAJE IN VARCHAR2) RETURN VARCHAR2;
-- FUNCIÓN QUE DEVUELVE EL ESTADO EN EL QUE SE ENCUENTRA UN ENVIO A MOVIL

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- MOVIL - VARCHAR2 - MOVIL DEL QUE SE QUIERE SABER SU ESTADO DE ENVIO
-- MENSAJE - VARCHAR2 - MENSAJE AL QUE PERTECENECE ESE ENVÍO

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

-- VALORES DEVUELTOS
-- VARCHAR2 - ESTADO DEL ENVIO

/*-----*/
/* PR_MENSAJES_USUARIO */
/*-----*/
PROCEDURE PR_MENSAJES_USUARIO(MENSAJES IN OUT COL_AR_MENSAJES, DNI IN VARCHAR2, ESTADO IN
NUMBER, QUERY IN VARCHAR2 DEFAULT NULL);
-- PROCEDIMIENTO QUE DEVUELVE TODOS LOS MENSAJES DE UN USUARIO

-- PARAMETROS DE ENTRADA/SALIDA OBLIGATORIOS
-- MENSAJES - COL_AR_MENSAJES - DONDE SE VA A DEVOLVER TODOS LOS MENSAJES

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- DNI - VARCHAR2 - DNI DEL USUARIO

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- ESTADO - VARCHAR2 - ESTADO DEL MENSAJE
-- QUERY - VARCHAR2 - CONSULTA

/*-----*/
/* PR_MENSAJES_MOVIL_USUARIO */
/*-----*/
PROCEDURE PR_MENSAJES_MOVIL_USUARIO(MENSAJES IN OUT COL_AR_MENSAJES, DNI IN VARCHAR2,
ESTADO IN NUMBER, QUERY IN VARCHAR2 DEFAULT NULL);
-- PROCEDIMIENTO QUE DEVUELVE TODOS LOS MENSAJES A MOVILES DE UN USUARIO

-- PARAMETROS DE ENTRADA/SALIDA OBLIGATORIOS
-- MENSAJES - COL_AR_MENSAJES - DONDE SE VA A DEVOLVER TODOS LOS MENSAJES

```

```
-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- DNI                - VARCHAR2          - DNI DEL USUARIO

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- ESTADO             - VARCHAR2          - ESTADO DEL MENSAJE
-- QUERY              - VARCHAR2          - CONSULTA

/*-----*/
/* PR_MENSAJE_CAMBIA_ESTADO                */
/*-----*/
PROCEDURE PR_MENSAJE_CAMBIA_ESTADO(EHASH IN VARCHAR2,EMAIL IN VARCHAR2,NUEVO_ESTADO IN
VARCHAR2,OBSERVACIONES IN VARCHAR2 DEFAULT NULL);
-- PROCEDIMIENTO QUE CAMBIA EL ESTADO DE UN MENSAJE

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- EHASH              - VARCHAR2          - CODIGO HASH DEL MENSAJE
-- EMAIL              - VARCHAR2          - EMAIL DEL USUARIO
-- NUEVO_ESTADO      - VARCHAR2          - ESTADO NUEVO

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- OBSERVACIONES     - VARCHAR2          - OBSERVACIONES PARA ESTE CAMBIO

/*-----*/
/* PR_MENSAJE_MVL_CAMBIA_ESTADO            */
/*-----*/
PROCEDURE PR_MENSAJE_MVL_CAMBIA_ESTADO(EHASH IN VARCHAR2,MOVIL IN VARCHAR2,NUEVO_ESTADO
IN VARCHAR2,OBSERVACIONES IN VARCHAR2 DEFAULT NULL);
-- PROCEDIMIENTO QUE CAMBIA EL ESTADO DE UN MENSAJE

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- EHASH              - VARCHAR2          - CODIGO HASH DEL MENSAJE
-- MOVIL              - VARCHAR2          - MOVIL DEL USUARIO
-- NUEVO_ESTADO      - VARCHAR2          - ESTADO NUEVO

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- OBSERVACIONES     - VARCHAR2          - OBSERVACIONES PARA ESTE CAMBIO

/*-----*/
/* PR_ENVIO_ESTADO_LEIDO                    */
/*-----*/
PROCEDURE PR_ENVIO_ESTADO_LEIDO(MENSAJE IN VARCHAR2, DNI IN VARCHAR2);
-- PROCEDIMIENTO QUE CAMBIA EL ESTADO DE UN ENVIO A LEIDO

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- MENSAJE           - VARCHAR2          - CODIGO HASH DEL MENSAJE
-- DNI                - VARCHAR2          - DNI DEL USUARIO

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

/*-----*/
/* PR_ENVIO_ESTADO_LEIDO_EMAIL              */
/*-----*/
PROCEDURE PR_ENVIO_ESTADO_LEIDO_EMAIL(MENSAJE IN VARCHAR2, EMAIL IN VARCHAR2);
-- PROCEDIMIENTO QUE CAMBIA EL ESTADO DE UN MENSAJE A LEIDO

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- MENSAJE           - VARCHAR2          - CODIGO HASH DEL MENSAJE
-- EMAIL             - VARCHAR2          - EMAIL DEL USUARIO

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

/*-----*/
/* PR_ENVIO_ESTADO_RECHAZADO                */
/*-----*/
PROCEDURE PR_ENVIO_ESTADO_RECHAZADO(MENSAJE IN VARCHAR2, DNI IN VARCHAR2);
-- PROCEDIMIENTO QUE CAMBIA EL ESTADO DE UN ENVIO A RECHAZADO

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- MENSAJE           - VARCHAR2          - CODIGO HASH DEL MENSAJE
-- DNI                - VARCHAR2          - DNI DEL USUARIO

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

/*-----*/
/* PR_ENVIO_ESTADO_RECHAZADO_MAIL          */
/*-----*/
```

```
/*-----*/
PROCEDURE PR_ENVIO_ESTADO_RECHAZADO_MAIL(MENSAJE IN VARCHAR2, EMAIL IN VARCHAR2);
-- PROCEDIMIENTO QUE CAMBIA EL ESTADO DE UN ENVIO A RECHAZADO

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- MENSAJE - VARCHAR2 - CODIGO HASH DEL MENSAJE
-- EMAIL - VARCHAR2 - EMAIL DEL USUARIO

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

/*-----*/
/* FU_ENVIO_ESTADO */
/*-----*/
FUNCTION FU_ENVIOS_ESTADO(MENSAJE IN VARCHAR2, DNI IN VARCHAR2) RETURN VARCHAR2;
-- PROCEDIMIENTO QUE DEVUELVE EL ESTADO DE UN ENVIO

-- PARAMETROS DE ENTRADA OBLIGATORIOS
-- MENSAJE - VARCHAR2 - CODIGO HASH DEL MENSAJE
-- DNI - VARCHAR2 - DNI DEL USUARIO

-- VALORES POR DEFECTO O VALORES OPCIONALES
-- NINGUNO

-- VALORES DEVUELTOS
-- VARCHAR2 - ESTADO DEL ENVIO
```

6 API Servicios Web

El Javadoc de la interfaz webservice de esta parte se puede encontrar dentro del directorio y un ejemplo de la misma en:

/KitAvisador/Documentos/javadoc

/KitAvisador/Ejemplos/Webservice

respectivamente.

7 Historia de versiones

VERSION	FECHA	AUTOR	DESCRIPCIÓN
1.0.0	19/08/2005	FJCV	Primera Versión @visor v1.0.0