Los detalles técnicos del servicio:

|  |  |
| --- | --- |
| **Nombre del servicio** | **CorreosOutlook** |
| **Formato para intercambio de datos** | **XML** |

PRIMER METODO:

|  |  |  |
| --- | --- | --- |
| **Operación** | POST | |
| **Nombre del método Java:** | **REGISTROCORREOSPAC** | Nuevo |
| **Operación:** | Síncrono (Request/Response) | |
| **Descripción de la operación:** | Método encargado de registrar los correos recibidos de la bandeja Outlook a una base datos oracle llamada TABLA RC. RECLAMOS, los archivos adjuntos serán de formato PDF, el procedimiento validara los archivos adjuntos que sea formato PDF y de tamaño Máximo 5MB serán registrado en el TABLA RC.ARCHIVOS, en caso no tenga estas condiciones el error será registrado en la tabla RC.COMENTARIO. Inicialmente antes de proceso mencionado el método realizara el registro en las tablas de transacciones de RC.BITACORA, RC.BITACORAADJUNTO. | |

#### DATOS DE ENTRADA (SE ENVIARA AL SERVICIO WEB DESDE STUDIO .NET)

1. **Cabecera HTTP (EL SERVICIO RECIBE LOS SIGUIENTE):**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre** | **Tipo** | **Longitud** | **Descripción** |
| Código | String | *-* | Identificador único generado por el tabla Reclamo. |
| Remitente | String | *-* | Identificador del cliente que enviara el correo. |
| Receptor | String | *-* | Identificador del buzón que recibe el correo. |
| Copiados | String | *-* | Identificador las usuarios copiados en el correo recibido. |
| Asunto | String | *-* | Descripción del asunto del correo recibido. |
| Cuerpo | String | *-* | Descripción del contenido del correo recibido. |
| FechaRecepcion | Date | *-* | Fecha de Recepción del Correo recibido. |
| Comentario | String | *-* | Comentarios de correos recibidos no cargados |
| ListaAdjunto | List |  | Lista de entidades adjuntos |
| TipoComentario | String |  | Validación Máximo Caracteres. |
| RutaArchivo | String |  | Ruta del file server donde se alojara el Archivo adjunto. |

**Lista de Entidad Adjunto:**

1. **Cabecera**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre** | **Tipo** | **Longitud** | **Descripción** |
| Nombre | String | - | Nombre Archivo Adjunto |
| Extensión | String | - | Extensión Archivo Adjunto |
| Comentario | String | - | Comentario Validación del adjunto |

#### DATOS DE SALIDA (RESPONSE)

1. **CUERPO: EL SERVICIO WE RESPONDERA A STUDIO VISUAL .NET LO SIGUIENTE:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre** | **Tipo** | **Longitud** | **Descripción** |
| CodigoRespuesta(Estado) | String | - | Código del mensaje de Exitoso y Errores. |
| CodigoError | String | - | proporciona mensaje del error (log WS) |
| IdReclamo | String | - | Identificador de Reclamo |

* Si codigoRespuesta = **“0”** **Exitoso Registrado** configurable en properties)
* Si codigoRespuesta = **“1”** **Interacción Incompleta**(configurable en properties)
* Si codigoRespuesta = **“2”** **Pendiente de Registro**(configurable en properties)
* Si codigoRespuesta = **“3”** **Cantidad Máximo Intentos** (3veces)(configurable en properties)

**ERRORES CONECTIVIDAD:**

* Si ocurrió algún error de TIMEOUT durante la invocación del método y terminar el proceso.
* Si ocurrió algún error de disponibilidad durante la invocación del método se

Para cualquier otro error inesperado y terminar con el proceso.

1. **Respuestas funcionales del método:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **codigoRespuesta** | **descripción Respuesta** | **Parte** |
| 0 | 0 | **Exitoso Registrado** | Body |
| 1 | 1 | **Interacción Incompleta** | Body |
| 2 | 2 | **Pendiente de Registro** | Body |
| 3 | 3 | **Cantidad Máximo Intentos** |  |

1. **Respuestas técnicas del método:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **código Error** | **Mensaje Error** | **Parte** |
| 1 | -1 | Error de disponibilidad en la DB [nombreBD] | Body |
| 2 | -2 | Error de TIMEOUT en el servicio: [nombreWS] | Body |
| 3 | -3 | Error de Disponibilidad en el servicio: [nombreWS] | Body |
| 4 | -4 | Ocurrió un error inesperado: [detalle] | Body |

TABLAS Y STORE PROCEDURE QUE INVOCARA EL SERVICIO WEB

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Esquema de BD** | **Tipo de objeto** | **Nombre de objeto** | **Descripción** | **Nuevo / Modificado / Eliminado / Reutilizado** |
| RC | Tabla | RECLAMO | Tabla historial de estados de interacciones en | Nuevo |
| RC | Tabla | ARCHIVO | Tabla de archivos ruta de adjuntos | Nuevo |
| RC | Tabla | COMENTARIO | Tabla de detalle de Error de solicitudes de registro | Nuevo |
| RC | Tabla | RECLAMOESTADO | Tabla de estado inicial de Registro de correos, luego son estados de cada interacción actualizada. | Nuevo |
| RC | Tabla | BITACORA | Tabla de historial de Registro total de correos Outlook. | Nuevo |
| RC | Tabla | BITACORAADJUNTO | Tabla de historial de correos con archivos adjuntos | Nuevo |
| RC | Tabla | BITACORARECHAZOS | Tabla que contendrá el código de tipo Rechazo | Nuevo |
| RC | **Paquete** | PKG\_PADMIN\_CORREOSPAC | Paquete que contiene los procedimientos de alquileres de PADMIN | Nuevo |
| RC | Procedimiento | SP.REGISTROBITACORA | Procedimiento que se encarga de o los registros de recibos del Outlook. | Nuevo |
| RC | Procedimiento | SP.REGISTROBITACORAADJUNTO | Procedimiento que se encarga de guardar el archivo adjuntos por cada correos recibidos del Outlook | Nuevo |
| RC | Procedimiento | SP.REGISTRORECLAMO | Procedimiento se encargara de registrarlo en tabla matriz de sistema PAC. | Nuevo |
| RC | Procedimiento | SP.REGISTROARCHIVO | Procedimiento que se encarga de guardar y agregar la ruta del File Server de los archivos Adjuntos. | Nuevo |
| RC | Procedimiento | SP.REGISTROCOMENTARIO | Procedimiento que se encarga de Registrar los comentarios de acuerdo al estado del proceso de registro a la tabla matriz Reclamos | Nuevo |
| RC | Procedimiento | SP.CUADRATURABUZONES | Procedimiento que se encarga de obtener la Lista de la cantidad correos obtenidos por cada buzón de Outlook. | Nuevo |
| RC | Procedimiento | SP.CUADRATURAINTERACCIONES | Procedimiento que se encargara de obtener la lista de correos registrados correctamente a la tabla matriz reclamos. | Nuevo |

CREATE OR REPLACE PACKAGE BODY RC.PKG\_PADMIN\_CORREOSPAC AS

// VA REGISTRAR LOS PARAMETROS ENVIADOS

PROCEDURE RegistrarBitacora(p\_CODIGO IN BITACORA.CODIGO%TYPE,

p\_REMITENTE IN BITACORA.REMITENTE%TYPE,

p\_RECEPTOR IN BITACORA.RECEPTOR%TYPE,

p\_COPIADOS IN BITACORA.COPIADOS%TYPE,

p\_ASUNTO IN BITACORA.ASUNTO%TYPE,

p\_CUERPO IN BITACORA.CUERPO%TYPE,

p\_FECHARECEPCION IN BITACORA.FECHARECEPCION%TYPE,

p\_COMENTARIO IN BITACORA.COMENTARIO%TYPE,

p\_TIPOCOMENTARIO IN BITACORA.TIPOCOMENTARIO%TYPE) IS

v\_IDBITACORA NUMBER;

BEGIN

SELECT SEQ\_BITACORA.NEXTVAL INTO v\_IDBITACORA from DUAL;

INSERT INTO BITACORA

(IDBITACORA,

CODIGO,

REMITENTE,

RECEPTOR,

COPIADOS,

ASUNTO,

CUERPO,

FECHARECEPCION,

FECHATRANSACCION,

COMENTARIO,

TIPOCOMENTARIO)

VALUES

(v\_IDBITACORA,

p\_CODIGO,

p\_REMITENTE,

p\_RECEPTOR,

p\_COPIADOS,

p\_ASUNTO,

p\_CUERPO,

p\_FECHARECEPCION,

sysdate,

p\_COMENTARIO,

p\_TIPOCOMENTARIO);

END;

PROCEDURE ActualizarBitacora(p\_IDBITACORA IN BITACORA.IDBITACORA%TYPE,

p\_IDRECLAMO IN BITACORA.IDRECLAMO%TYPE,

p\_CODIGORECHAZO IN BITACORA.CODIGORECHAZO%TYPE,

p\_TRANSACCION IN BITACORA.TRANSACCION%TYPE) IS

BEGIN

UPDATE BITACORA

SET IDRECLAMO = p\_IDRECLAMO,

CODIGORECHAZO = p\_CODIGORECHAZO,

TRANSACCION = p\_TRANSACCION

WHERE IDBITACORA = p\_IDBITACORA;

END;

// REGISTRARA LOS CORREOS CON ARCHIVOS ADJUNTOS CON PDF ENVIADO POR EL VISUAL STUDIO .NET

PROCEDURE RegistrarBitacoraAdjunto(p\_IDBITACORA IN BITACORAADJUNTO.IDBITACORA%TYPE,

p\_NOMBRE IN BITACORAADJUNTO.NOMBRE%TYPE,

p\_EXTENSION IN BITACORAADJUNTO.EXTENSION%TYPE,

p\_NOMBREPAC IN BITACORAADJUNTO.NOMBREPAC%TYPE,

p\_ARCHIVO IN BITACORAADJUNTO.ARCHIVO%TYPE,

p\_COMENTARIO IN BITACORAADJUNTO.COMENTARIO%TYPE) IS

V\_IDBITACORAADJUNTO NUMBER;

BEGIN

SELECT SEQ\_BITACORAADJUNTO.NEXTVAL INTO V\_IDBITACORAADJUNTO FROM DUAL;

INSERT INTO BITACORAADJUNTO

(IDBITACORAADJUNTO,

IDBITACORA,

NOMBRE,

EXTENSION,

NOMBREPAC,

ARCHIVO,

COMENTARIO)

VALUES

(V\_IDBITACORAADJUNTO,

p\_IDBITACORA,

p\_NOMBRE,

p\_EXTENSION,

p\_NOMBREPAC,

p\_ARCHIVO,

p\_COMENTARIO);

END;

//TERMINA DE REGISTRAR EN LA TABLAS BITACORA SIGUE EL MISMO ACA

PROCEDURE RegistrarReclamo(p\_FECHARECEPCION IN RECLAMO.FECHARECEPCION%TYPE,

p\_PROBLEMA IN RECLAMO.PROBLEMA%TYPE,

p\_REMITENTE IN RECLAMO.REMITENTE%TYPE) IS

v\_IDRECLAMO RECLAMO.IDRECLAMO%TYPE;

v\_NUMERORECLAMO RECLAMO.NUMERORECLAMO%TYPE;

v\_IDTIPOINTERACCION RECLAMO.IDTIPOINTERACCION%TYPE;

v\_IDESTADO RECLAMO.IDESTADO%TYPE;

v\_IDTOPICO RECLAMO.IDTOPICO%TYPE;

v\_IDVIAREQUERIMIENTO RECLAMO.IDVIAREQUERIMIENTO%TYPE;

v\_IDTEMA RECLAMO.IDTEMA%TYPE;

v\_IDAREAINGRESO RECLAMO.IDAREAINGRESO%TYPE;

v\_IDCANALINGRESO RECLAMO.IDCANALINGRESO%TYPE;

v\_IDAGENCIA RECLAMO.IDAGENCIA%TYPE;

v\_USUARIOREGISTRA RECLAMO.USUARIOREGISTRA%TYPE;

v\_USUARIO VARCHAR2(80);

BEGIN

v\_NUMERORECLAMO := 'SN00000';

v\_IDTIPOINTERACCION := '';

v\_IDTOPICO := 0;

v\_IDTEMA := '';

v\_IDCANALINGRESO := 'CNI014';

v\_IDESTADO := 'ESR001';

v\_IDVIAREQUERIMIENTO := 'VRE009';

v\_IDAGENCIA := 18;

v\_IDAREAINGRESO := 9;

v\_USUARIO := 'OUTLOOK';

SELECT SEQ\_RECLAMO.NEXTVAL INTO v\_IDRECLAMO FROM DUAL;

SELECT IDUSUARIO

INTO v\_USUARIOREGISTRA

FROM RC.USUARIO

WHERE UPPER(USUARIO) = v\_USUARIO;

INSERT INTO RECLAMO

(IDRECLAMO,

NUMERORECLAMO,

IDESTADO,

FECHAREGISTRO,

PROBLEMA,

IDTOPICO,

IDTIPOINTERACCION,

IDVIAREQUERIMIENTO,

IDTEMA,

IDAREAINGRESO,

IDCANALINGRESO,

USUARIOREGISTRA,

FECHARECEPCION,

IDAGENCIA,

REMITENTE)

VALUES

(v\_IDRECLAMO,

v\_NUMERORECLAMO,

v\_IDESTADO,

sysdate,

p\_PROBLEMA,

v\_IDTOPICO,

v\_IDTIPOINTERACCION,

v\_IDVIAREQUERIMIENTO,

v\_IDTEMA,

v\_IDAREAINGRESO,

v\_IDCANALINGRESO,

v\_USUARIOREGISTRA,

p\_FECHARECEPCION,

v\_IDAGENCIA,

p\_REMITENTE);

PKG\_RECLAMO.ReclamoEstadoRegistrar(v\_IDRECLAMO, null, null, v\_IDESTADO, sysdate, null,

null, sysdate, v\_USUARIOREGISTRA, v\_USUARIOREGISTRA);

END;

//IGUALMENTE REGISTRA ACA CUANDO TIENE ARCHIVOS ADJUNTOS

PROCEDURE RegistrarArchivo(P\_IDENTIDAD IN ARCHIVO.IDENTIDAD%TYPE,

P\_DESCRIPCION IN ARCHIVO.DESCRIPCION%TYPE,

P\_RUTAARCHIVO IN ARCHIVO.RUTAARCHIVO%TYPE,

P\_NOMBREARCHIVO IN ARCHIVO.NOMBREARCHIVO%TYPE,

P\_EXTENSION IN ARCHIVO.EXTENSION%TYPE) IS

V\_IDARCHIVO ARCHIVO.IDARCHIVO%TYPE;

V\_IDTIPOARCHIVO ARCHIVO.IDTIPOARCHIVO%TYPE;

V\_USUARIOREGISTRA ARCHIVO.USUARIOREGISTRA%TYPE;

BEGIN

SELECT SEQ\_ARCHIVO.NEXTVAL INTO V\_IDARCHIVO FROM DUAL;

V\_IDTIPOARCHIVO := 'TAR013';

INSERT INTO ARCHIVO

(IDARCHIVO,

IDENTIDAD,

DESCRIPCION,

RUTAARCHIVO,

NOMBREARCHIVO,

EXTENSION,

IDTIPOARCHIVO,

USUARIOREGISTRA)

VALUES

(V\_IDARCHIVO,

P\_IDENTIDAD,

P\_DESCRIPCION,

P\_RUTAARCHIVO,

P\_NOMBREARCHIVO,

P\_EXTENSION,

V\_IDTIPOARCHIVO,

V\_USUARIOREGISTRA);

END;

// ESTE INSERTA UN COMENTARIO DE ERROR SI NO REGISTRO EN LA TABLA DE RECLAMO QUE TAMBIEN ESTA EL PARAMETRO

PROCEDURE RegistrarComentario(P\_TEXTO IN COMENTARIO.TEXTO%TYPE,

P\_IDRECLAMO IN COMENTARIO.IDRECLAMO%TYPE) IS

V\_IDCOMENTARIO COMENTARIO.IDCOMENTARIO%TYPE;

V\_USUARIOREGISTRA COMENTARIO.USUARIOREGISTRA%TYPE;

BEGIN

SELECT SEQ\_COMENTARIORECLAMO.NEXTVAL INTO V\_IDCOMENTARIO FROM DUAL;

V\_USUARIOREGISTRA := 0;

INSERT INTO COMENTARIO

(IDCOMENTARIO, TEXTO, IDRECLAMO, USUARIOREGISTRA)

VALUES

(V\_IDCOMENTARIO, P\_TEXTO, P\_IDRECLAMO, V\_USUARIOREGISTRA);

END;

PROCEDURE RegistrarReclamoConsolidar(p\_IDRECLAMO IN VARCHAR2,

p\_IDRECLAMOPADRE IN RECLAMOCONSOLIDAR.IDRECLAMOPADRE%TYPE,

p\_USUARIOREGISTRA IN RECLAMOCONSOLIDAR.USUARIOREGISTRA%TYPE,

p\_RESULTADO OUT NUMBER) IS

v\_IDTIPOENTIDAD ARCHIVO.IDTIPOENTIDAD%TYPE;

v\_IDESTADO RECLAMO.IDESTADO%TYPE;

BEGIN

p\_RESULTADO := 1;

v\_IDTIPOENTIDAD := 'TEN001';

v\_IDESTADO := 'ESR004';

*-- INSERTAR RECLAMOCONSOLIDAR --*

INSERT INTO RECLAMOCONSOLIDAR

(IDRECLAMOCONSOLIDAR, IDRECLAMO, IDRECLAMOPADRE, USUARIOREGISTRA)

SELECT SEQ\_RECLAMOCONSOLIDAR.NEXTVAL,

TO\_NUMBER(IDSPLIT),

p\_IDRECLAMOPADRE,

p\_USUARIOREGISTRA

FROM TABLE(PKG\_REPORTE.FCSPLIT(p\_IDRECLAMO, ','));

*-- INSERTAR ARCHIVO --*

INSERT INTO ARCHIVO

(IDARCHIVO,

IDTIPOENTIDAD,

IDENTIDAD,

DESCRIPCION,

IDTIPOARCHIVO,

RUTAARCHIVO,

USUARIOREGISTRA,

NOMBREARCHIVO,

EXTENSION)

SELECT SEQ\_ARCHIVO.NEXTVAL,

v\_IDTIPOENTIDAD,

p\_IDRECLAMOPADRE,

A.DESCRIPCION,

A.IDTIPOARCHIVO,

A.RUTAARCHIVO,

p\_USUARIOREGISTRA,

A.NOMBREARCHIVO,

A.EXTENSION

FROM ARCHIVO A

INNER JOIN (SELECT IDSPLIT AS IDRECLAMO

FROM TABLE(PKG\_REPORTE.FCSPLIT(p\_IDRECLAMO, ','))) T

ON A.IDENTIDAD = TO\_NUMBER(T.IDRECLAMO);

*-- INSERTAR COMENTARIO --*

INSERT INTO COMENTARIO

(IDCOMENTARIO, TEXTO, IDRECLAMO, USUARIOREGISTRA)

SELECT SEQ\_COMENTARIORECLAMO.NEXTVAl,

C.TEXTO,

p\_IDRECLAMOPADRE,

p\_USUARIOREGISTRA

FROM COMENTARIO C

INNER JOIN (SELECT IDSPLIT AS IDRECLAMO

FROM TABLE(PKG\_REPORTE.FCSPLIT(p\_IDRECLAMO, ','))) T

ON C.IDRECLAMO = TO\_NUMBER(T.IDRECLAMO);

*-- ACTUALIZAR COMENTARIO --*

UPDATE COMENTARIO C

SET C.TEXTO = C.TEXTO || '-' || TO\_CHAR(p\_IDRECLAMOPADRE),

C.FECHAMODIFICA = SYSDATE,

C.USUARIOMODIFICA = p\_USUARIOREGISTRA

WHERE EXISTS (SELECT 1

FROM (SELECT IDSPLIT AS IDRECLAMO

FROM TABLE(PKG\_REPORTE.FCSPLIT(p\_IDRECLAMO, ','))) T

WHERE C.IDRECLAMO = TO\_NUMBER(T.IDRECLAMO));

*-- ACTUALIZAR RECLAMO --*

UPDATE RECLAMO R

SET R.IDESTADO = v\_IDESTADO,

R.FECHAMODIFICA = SYSDATE,

R.USUARIOMODIFICA = p\_USUARIOREGISTRA

WHERE EXISTS (SELECT 1

FROM (SELECT IDSPLIT AS IDRECLAMO

FROM TABLE(PKG\_REPORTE.FCSPLIT(p\_IDRECLAMO, ','))) T

WHERE R.IDRECLAMO = TO\_NUMBER(T.IDRECLAMO));

COMMIT;

EXCEPTION

WHEN OTHERS THEN

ROLLBACK;

p\_RESULTADO := 0;

END;

PROCEDURE ConsultarInteraccionesOutlook(p\_NUMERORECLAMO IN RECLAMO.NUMERORECLAMO%TYPE,

p\_IDTIPOINTERACCION IN VARCHAR2,

p\_FECHAINICIAL IN DATE DEFAULT NULL,

p\_FECHAFINAL IN DATE DEFAULT NULL,

p\_TIPOFECHA in VARCHAR2,

p\_NumPagina IN NUMBER,

p\_NumRegxPagina IN NUMBER,

p\_Total OUT NUMBER,

p\_TRCNSCC OUT TrcnRec) IS

v\_RegInicio NUMBER;

v\_RegFin NUMBER;

v\_CONTADORTIPOREQUERIMIENTO number;

v\_ESTADO VARCHAR2(6);

v\_USUARIOREGISTRA RECLAMO.USUARIOREGISTRA%TYPE;

v\_USUARIO VARCHAR2(80);

BEGIN

v\_RegFin := p\_NumPagina \* p\_NumRegxPagina;

v\_RegInicio := v\_RegFin - p\_NumRegxPagina + 1;

v\_ESTADO := 'ESR001';

v\_USUARIO := 'OUTLOOK';

SELECT COUNT(IDSPLIT)

INTO v\_CONTADORTIPOREQUERIMIENTO

FROM TABLE(RC.PKG\_REPORTE.FCSPLIT(p\_IDTIPOINTERACCION, ',')); *-- tipo interacci??n*

SELECT IDUSUARIO

INTO v\_USUARIOREGISTRA

FROM RC.USUARIO

WHERE UPPER(USUARIO) = v\_USUARIO;

SELECT COUNT(\*)

INTO p\_Total

FROM RECLAMO REC

WHERE (p\_NUMERORECLAMO IS NULL OR UPPER(REC.NUMERORECLAMO) like

'%' || UPPER(p\_NUMERORECLAMO) || '%')

AND (p\_IDTIPOINTERACCION IS NULL OR

(v\_ContadorTipoRequerimiento > 1) OR

REC.IDTIPOINTERACCION IN

(select idSplit

from table(RC.PKG\_REPORTE.fcSplit(p\_IDTIPOINTERACCION, ','))))

AND (p\_TipoFecha IS NULL OR

(p\_TipoFecha = '001' AND

((p\_FechaInicial IS NULL OR

TRUNC(REC.FECHAREGISTRO) >= TRUNC(p\_FechaInicial)) AND

(p\_FechaFinal IS NULL OR

TRUNC(REC.FECHAREGISTRO) <= TRUNC(p\_FechaFinal)))) OR

(p\_TipoFecha = '002' AND

((p\_FechaInicial IS NULL OR

TRUNC(REC.FECHARECEPCION) >= TRUNC(p\_FechaInicial)) AND

(p\_FechaFinal IS NULL OR

TRUNC(REC.FECHARECEPCION) <= TRUNC(p\_FechaFinal)))))

AND (REC.IDESTADO = v\_ESTADO)

AND (REC.USUARIOREGISTRA = v\_USUARIOREGISTRA);

OPEN p\_TRCNSCC FOR

SELECT REC.IDRECLAMO,

REC.NUMERORECLAMO,

REC.FECHAREGISTRO,

REC.FECHARECEPCION,

REC.IDTIPOINTERACCION,

REC.REMITENTE,

AG.DESCRIPCION AS AGENCIA,

ARE.DESCRIPCION AS desAreaIngreso,

RC.PKG\_PARAMETRO.sf\_Descripcion(REC.IDCANALINGRESO) AS DescCanalIngreso,

RC.PKG\_PARAMETRO.sf\_Descripcion(REC.IDVIAREQUERIMIENTO) AS DescViaRequerimiento,

USUREG.NOMBRECOMPLETO AS USUARIOREGISTRA,

RC.PKG\_PARAMETRO.sf\_Descripcion(REC.IDESTADO) AS DESESTADORECLAMO

FROM (SELECT S.NUMFILA, S.IDRECLAMO

FROM (SELECT ROW\_NUMBER() OVER(ORDER BY REC.FECHAREGISTRO DESC) AS NUMFILA,

REC.IDRECLAMO

FROM RC.RECLAMO REC

WHERE (p\_NUMERORECLAMO IS NULL OR

UPPER(REC.NUMERORECLAMO) like

'%' || UPPER(p\_NUMERORECLAMO) || '%')

AND (p\_IDTIPOINTERACCION IS NULL OR

(v\_ContadorTipoRequerimiento > 1) OR

REC.IDTIPOINTERACCION IN

(select idSplit

from table(RC.PKG\_REPORTE.fcSplit(p\_IDTIPOINTERACCION,

','))))

AND (p\_TipoFecha IS NULL OR

(p\_TipoFecha = '001' AND

((p\_FechaInicial IS NULL OR

TRUNC(REC.FECHAREGISTRO) >=

TRUNC(p\_FechaInicial)) AND

(p\_FechaFinal IS NULL OR TRUNC(REC.FECHAREGISTRO) <=

TRUNC(p\_FechaFinal)))) OR

(p\_TipoFecha = '002' AND

((p\_FechaInicial IS NULL OR

TRUNC(REC.FECHARECEPCION) >=

TRUNC(p\_FechaInicial)) AND

(p\_FechaFinal IS NULL OR TRUNC(REC.FECHARECEPCION) <=

TRUNC(p\_FechaFinal)))))

AND (REC.IDESTADO = v\_ESTADO)

AND (REC.USUARIOREGISTRA = v\_USUARIOREGISTRA)) S

WHERE S.NUMFILA BETWEEN v\_RegInicio AND v\_RegFin) O

INNER JOIN RC.RECLAMO REC

ON O.IDRECLAMO = REC.IDRECLAMO

LEFT JOIN RC.AGENCIA AG

ON REC.IDAGENCIA = AG.IDAGENCIA

LEFT JOIN RC.AREA ARE

on REC.IDAREAINGRESO = ARE.IDAREA

INNER JOIN RC.USUARIO USUREG

ON REC.USUARIOREGISTRA = USUREG.IDUSUARIO

ORDER BY O.NUMFILA ASC;

END;

PROCEDURE BuscarInteraccionesOutlookPorR(p\_IDRECLAMO IN RECLAMO.IDRECLAMO%TYPE,

p\_TRCNSCC OUT TrcnRec) IS

BEGIN

OPEN p\_TRCNSCC FOR

SELECT REC.IDRECLAMO,

REC.NUMERORECLAMO,

REC.FECHAREGISTRO,

REC.FECHARECEPCION,

REC.IDTIPOINTERACCION,

REC.REMITENTE,

AG.DESCRIPCION AS AGENCIA,

ARE.DESCRIPCION AS desAreaIngreso,

RC.PKG\_PARAMETRO.sf\_Descripcion(REC.IDCANALINGRESO) AS DescCanalIngreso,

RC.PKG\_PARAMETRO.sf\_Descripcion(REC.IDVIAREQUERIMIENTO) AS DescViaRequerimiento,

USUREG.NOMBRECOMPLETO AS USUARIOREGISTRA,

RC.PKG\_PARAMETRO.sf\_Descripcion(REC.IDESTADO) AS DESESTADORECLAMO

FROM RC.RECLAMO REC

LEFT JOIN RC.AGENCIA AG

ON REC.IDAGENCIA = AG.IDAGENCIA

LEFT JOIN RC.AREA ARE

on REC.IDAREAINGRESO = ARE.IDAREA

INNER JOIN RC.USUARIO USUREG

ON REC.USUARIOREGISTRA = USUREG.IDUSUARIO

WHERE REC.IDRECLAMO = p\_IDRECLAMO;

END;

END PKG\_PADMIN\_CORREOSPAC;