

REPORTE CAMBIO DE NOMBRE

FUENTE: /var/www/html/qareportes/app/Http/Controllers/Cloud/ReportController.php

1) Nombre del reporte

- Nombre: Reporte Cambio de Precio
- Código/Alias: report_cambio_precio
- Responsable: Equipo de Reportes Cloud
- Propósito:

2) Alcance temporal

- Campo(s) de fecha utilizados (base):
 - first_date
 - second_date
 - created_date
- Tipo de rango: Personalizado (entre firstDate -> Primer día del mes y secondDate -> último día del mes)
- Inclusión de horas (sí/no). Si sí:
 - first_date [00:00:00]
 - second_date [23:59:59]
- Zona horaria: Servidor (igual a sistema)

3) Origen de datos (tablas/APIs)

3.1 Conexión / Base: Empresarial

- Tabla / Recurso: emp_ordenservicio
- Alias (si aplica): no hay
- Campos utilizados (clave → descripción):
 - ose_estadoPago → Estado de pago de la guia
 - ose_id → ID de la GUIA
 - ose_fechaanulado → Fecha de anulacion
 - ose_fhpreferenciapartida → Fecha y hora de partida
 - ose_serieguiacliente → Serie Cliente
 - ose_nroguiacliente → Numero guia
 - ose_termorigenatencion → Terminal Origen
 - ose_termdestinoentrega → Terminal Destino
 - ose_remiteempresa → Nombre del remitente
 - ose_destinaempresa → Nombre Destinatario
 - '0' → Contenido total
 - '' → Contenido
 - '0' → Peso
 - '0' → volumen
 - ose_servpagotipo → Forma de pago
 - ose_montototal → Monto
 - ose_precio_cambioprecio → Cambio de precio
 - diferencia → Es el resultado de la resta entre ose_montototal y ose_precio_cambioprecio
 - ose_tipoCP → Tipo comprobante
 - ose_serieguiacliente → Serie de la guia
 - ose_nroCP → Numero comprobante
 - ose_entregadopersonadni →Codigo de Entrega
 - ose_entregadopersonadni → Mensaje de codigo de Entrega
 - ose_entregadofecha a→ Fecha de entrega
 - ose_destina_direccion_entrega → Estado guia (Entregado en Agencia)
 - ose_movilidad_reparto → Indica si tiene reparto o no
 - ose_guia_remitente → null
 - ose_user_cambioprecio → Usuario que realizó el cambio de precio
 - ose_user_cambioprecio → Terminal del usuario que cambió precio
 - ose_fecha_cambioprecio → Fecha del cambio de precio
 - ose_fecha_cambioprecio → Hora del cambio de precio
 - ose_observacion_cambioprecio → Observacion del cambio de precio
 - usercreaid → Usuario que digite o creo la guia

- proforma_user → valor: NULL / No asignado
- ose_remitecontactofono → Contacto con cliente

- Condiciones (WHERE) aplicadas:
 - ose_estado = 1
 - eliminado = 0
 - ose_estadoPago != 'DA'
 - ose_estado_solicitudprecio = 1
 - ose_fecha_cambioprecio → Primer y ultimo dia del mes (first_date, second_date)

- Tipo de unión con otras fuentes (JOIN y llave): No aplica
- Observaciones (ej. particiones, índices): No hay particiones o índices especificados

3.2 Conexión / Base: empresarial

- Tabla / Recurso: emp_producto

- Alias (si aplica): No aplica

- Condiciones (WHERE) aplicadas: No aplica

- Tipo de unión con otras fuentes (JOIN y llave): No aplica

- Campos utilizados (clave → descripción):
 - pro_producto → tipo de producto
 - pro_id → ID del producto

Observaciones: Trae solo tipo del producto y su ID

3.3 Conexión / Base: empresarial

- Tabla / Recurso: emp_atributos

- Alias (si aplica): No aplica

- Condiciones (WHERE) aplicadas:

- atr_idpadre = 90

- Tipo de unión con otras fuentes (JOIN y llave): No aplica
- Campos utilizados (clave → descripción):
 - atr_nombre → Nombre del atributo
 - atr_valor → Valor del atributo

Observaciones: Trae solo tipo El nombre del atributo su valor

3.4 Conexión / Base: empresarial

- Tabla / Recurso: emp_usuario
- Alias (si aplica): No aplica
- Condiciones (WHERE) aplicadas:
 - whereIn('usr_id', \$strvalusr)
 - Se tiene un chunk que procesa los datos en cantidades de 1000
 - Solo trae a los usuarios en el cual su ID coincida dentro del grupo de 1000
- Tipo de unión con otras fuentes (JOIN y llave): No aplica
- Campos utilizados (clave → descripción):
 - usr_id → ID de usuario
 - usr_alias → Nombre del usuario
 - usr_terminalid → Terminal del usuario

Observaciones: Trae solo solo los usuario que coincidan dentro del chunk o grupo de 1000

3.5 Conexión / Base: empresarial

- Tabla / Recurso: emp_os_detalle
- Alias (si aplica): No aplica
- Condiciones (WHERE) aplicadas:
 - osd_eliminado = 0

- trae solo las OS que no hagay sido eliminadas
- whereIn('osd_osid', \$valdet)
 - Solo trae las osd_osid que se encuentren en el chunk de 1000
- Tipo de unión con otras fuentes (JOIN y llave): No aplica
- Campos utilizados (clave → descripción):
 - osd_unidad → Cantidad de paquetes
 - osd_osid → ID de la Guia
 - osd_descripcion → Descripción del producto (tipo producto)
 - osd_peso → Peso del producto
 - osd_volumen → Volumen del paquete

Observaciones:

3.6 Conexión / Base: empresarial

- Tabla / Recurso: emp_persona
- Alias (si aplica): No aplica
- Condiciones (WHERE) aplicadas:
 - whereIn('per_nrodocumento', \$strvalper)
 - Solo trae las per_nrodocumento que se encuentren en el chunk de 1000
- Tipo de unión con otras fuentes (JOIN y llave): No aplica
- Campos utilizados (clave → descripción):
 - per_nombrezsoc → Nombre
 - per_apellidopaterno → Apellido paterno
 - per_apellidomaterno → Apellido materno
 - per_nrodocumento → Documento de indentidad

Observaciones:

3.7 Conexión / Base: empresarial

- Tabla / Recurso: emp_cambio_precio

- Alias (si aplica): No aplica
- Condiciones (WHERE) aplicadas:
 - estado = 1
 - whereIn('ose_id', \$valdet)
 - Solo trae los registros que tengan el mismo ose_id que se encuentren en el chunk de 1000
- Tipo de unión con otras fuentes (JOIN y llave): No aplica
- Campos utilizados (clave → descripción):
 - urlimagen → Imagen cambio de precio
 - ose_id → ID registro cambio de precio

Observaciones:

3.8 Conexión / Base: empresarial

- Tabla / Recurso: emp_incidencias
- Alias (si aplica): No aplica
- Condiciones (WHERE) aplicadas:
 - inc_tipo = EDITAR PRECIO
 - whereIn('inc_idos', \$valinc)
 - Solo trae los registros que tengan el inc_idos que se encuentren en el chunk de 1000
 - Tambien hay una funcion anonima dentro de un where que filtra los registros que tengan el campo "eliminado" con valor null ó 0
- Tipo de unión con otras fuentes (JOIN y llave): No aplica
- Campos utilizados (clave → descripción):
 - inc_idos → ID de la Guia con incidencia
 - inc_tipo → Tipo de incidencia

Observaciones:

3.9 Conexión / Base: empresarial

- Tabla / Recurso: emp_soporte
- Alias (si aplica): No aplica
- Condiciones (WHERE) aplicadas:
 - where("sop_tipo", '=', 'editar_volumen_pro')
 - whereIn('sop_oseid', \$valinc)
 - Solo trae los registros que tengan el la misma Orden de servicio(sop_oseid) que se encuentren en el chunk de 1000
- Tipo de unión con otras fuentes (JOIN y llave): No aplica
- Campos utilizados (clave → descripción):
 - sop_anterior → Tipo de Pago anterior al cambio
 - sop_oseid → Orden de servicio

Observaciones:

4) Filtros globales del reporte

- Inclusiones (must-have):
 - ->where("ose_estado", "=", 1) → Solo órdenes activas.
 - ->where("ose_estado_solicitudprecio", "=", 1) → Solo órdenes con solicitud de precio habilitada.
 - ->whereBetween("ose_fecha_cambioprecio", [\$first_date, \$second_date]) → Solo registros dentro del rango de fechas.
- Exclusiones (reglas de descarte):
 - ->where("eliminado", "=", 0) → descarta eliminados.
 - ->where("ose_estadoPago", "!=", 'DA') → descarta órdenes con estado de pago "DA".
- Reglas por estado / habilitado:
 - El reporte solo toma órdenes vigentes/habilitadas (ose_estado = 1).
 - El cambio de precio debe estar activo (estado = '1' en emp_cambio_precio).

- Reglas para plataformas (ejemplo: "REGISTRA" → "ENVIA YA")
- Parámetros externos (si el usuario puede filtrarlo):
 - Rango de fecha → \$first_date y \$second_date

5) Transformaciones y Reglas de negocio

- Derivaciones de campos (cómo se calculan):
 - diferencia: Deriva de la resta entre los campos ose_montototal - ose_precio_cambioprecio
 - DATE_FORMAT(ose_fecha_cambioprecio, '%Y-%m-%d') as fhcambioprecio → transforma la fecha a formato YYYY-MM-DD
 - DATE_FORMAT(ose_fecha_cambioprecio, '%H:%i:%s') as hrcambioprecio → extrae solo la hora.
 - '0' as contenidototal, '0' as peso, '0' as volumen → inicializa siempre con 0.
 - '' as contenido → siempre vacío.
- Mapeos de estado:
 - ose_estado = 1 → mapea que solo se consideren órdenes activas.
 - ose_estadoPago != 'DA' → descarta un estado de pago específico (probablemente "Desaprobado" o similar).
 - ose_estado_solicitudprecio = 1 → asegura que esté aprobado para cambios de precio.
 - En otras partes (como incidencias), se aplica estado = '1' que mapea a "activo/habilitado".
- Reglas de validación (p.ej., sólo registros validados):
 - eliminado = 0 o whereNull/eliminado → garantiza que no entren registros eliminados.
 - Validación de rango de fechas con whereBetween("ose_fecha_cambioprecio", [\$first_date, \$second_date]) → asegura que solo se tomen registros dentro del periodo válido para el reporte.
 - En caso de error (catch), igual se inserta un log en emp_reportes_validation con status = 0.

- Reglas condicionales (si X entonces Y): Si el mes actual es enero o febrero → se amplía el rango incluyendo diciembre del año anterior

6) Estructura de salida

- Tabla/archivo destino: emp_reportes_validation
- Particionado / sufijo: No aplica
- Clave(s) primaria(s) o únicas: No aplica
- Columnas del reporte (nombre → tipo → descripción):
 - report→VARCHAR → Nombre del reporte generado
 - first_date→DATE→Fecha de inicio del rango de consulta
 - second_date→DATE→Fecha de fin del rango de consulta
 - created_date→DATETIME→Fecha y hora en que se generó el log de validación
 - status→INT / TINYINT→Estado de la ejecución (1 = OK, 0 = error)
- Ordenamiento por defecto: No aplica orderBy

7) Frecuencia y operación

- Frecuencia de actualización: Posible frecuencia: ejecución programada por CRON (cada día/mes)
- Ventana que recalcula (días/meses): Si es Enero o Febrero toma ambos meses. Pero a partir de Marzo comienza a tomar desde dos meses atrás.
- Tamaño de lote / paginado (chunking): array_chunk(..., 1000) → paginado por 1000 registros para las tablas:
 - emp_usuario
 - emp_persona
- Tolerancia a fallos / reintentos:
 - try/ catch
 - Si la consulta falla, igual inserta un registro en emp_reportes_validation con

status = 0

- Tiempos de ejecución esperados: No especifica

8) Calidad y controles

- Validaciones previas/post: Antes de insertar en la tabla emp_reportes_validation se valida que los datos de las tablas, pasen retornen true:
 - emp_ordenservicio
 - emp_producto
 - emp_atributos
 - emp_usuario
 - emp_os_detalle
 - emp_persona
 - emp_cambio_precio
 - emp_incidencias
 - emp_soporte
- Controles de duplicados: No aplica
- Métricas de completitud (qué se monitorea):
 - Se monitorea si el proceso realmente generó información (count(\$response_service) > 0).
 - También se marca con status = 1 o status = 0 según éxito o fallo, lo que sirve como métrica de calidad de ejecución.
 - Se guarda el rango de fechas (first_date, second_date) como evidencia del periodo cubierto.

9) Dependencias externas

- APIs / servicios y endpoints: No consume API externamente. EL reporte depende de una función interna llamada reportCambioPrecio(\$first_date, \$second_date) que tiene los parámetros de fecha.
- Autenticación / headers: No aplica
- Mapeos y contratos de datos:
 - "report" → Identificador del reporte

- "first_date" → Fecha inicial del rango procesado
- "second_date" → Fecha final del rango procesado
- "created_date" → date("Y-m-d H:i:s"),// Fecha/hora de la ejecución
- "status" => 1 | 0 → Estado del proceso (1=ok, 0=falla)

10) Historial de cambios

2025-09-15 16:42 - Elian Franco Arroyo Roda - Documentación inicial

Revisión #1

Creado 2025-10-24 15:25:16 -05 por Elian

Actualizado 2025-10-24 15:25:34 -05 por Elian