

REPORTE CENTRO COSTO DHL 2

Fuente: /var/www/html/qareportes/app/Http/Controllers/Cloud/CentroCostosController.php

1) Nombre del reporte

- Nombre: Reporte Centro Costo DHL 2
- Código / Alias: report_centro_costo_dhl_2
- Responsable: Equipo de Reportes Cloud
- Propósito (1 línea):

2) Alcance temporal

- Campo(s) de fecha utilizados: first_date, second_date
- Tipo de rango (diario / mensual / personalizado): Mensual (se recorren los meses del año actual y del año anterior)
- Inclusión de horas (sí/no).: Sí [00:00:00 - 23:59:59]
- Zona horaria: La del servidor

3) Origen de datos (tablas/APIs)

3.1 Conexión / Base: EMPRESARIAL

- Tabla / Recurso: emp_comppago_dhl
- Alias (si aplica): No aplica
- Campos utilizados (clave → descripción):
 - ter_id → Identificador del tercero asociado al envío.
 - ter_nombre → Nombre del tercero o cliente.
 - cop_montoTotal → Monto total del comprobante DHL.
 - cop_id → Identificador único del comprobante.

- cop_fechaemision → Fecha de emisión del comprobante.
- Condiciones (WHERE) aplicadas:
 - eliminado = 0
 - cop_fechaemision BETWEEN {first_date} 00:00:00 AND {second_date} 23:59:59
- Tipo de unión con otras fuentes (JOIN y llave): No aplica (consulta directa a tabla única).
- Observaciones (ej. particiones, índices):
 - Se realiza una suma total (SUM(cop_montoTotal)) y conteo (COUNT(cop_id)) para obtener el monto total y cantidad de comprobantes emitidos en el rango.
 - El resultado alimenta los campos ventas y qty_envios del reporte.

3.2 Conexión / Base: CAPACITACION

- Tabla / Recurso: tabSolicitud de Pagos
- Alias (si aplica): No aplica
- Campos utilizados (clave → descripción):
 - monto → Monto del pago realizado.
 - currency → Moneda del registro (USD o PEN).
 - proveedor → Nombre del proveedor (filtrado por “DHL EXPRESS PERÚ S.A.C.”).
 - mes_servicio_dhl → Mes del servicio facturado.
 - año_del_servicio → Año del servicio facturado.
 - concepto → Concepto del servicio (“Servicio Aéreo”).
 - fecha → Fecha del registro del pago.
- Condiciones (WHERE) aplicadas:
 - proveedor = 'DHL EXPRESS PERÚ S.A.C.'
 - concepto = 'Servicio Aereo'
 - mes_servicio_dhl = {mes_en_español}
 - año_del_servicio = {año}
- Tipo de unión con otras fuentes (JOIN y llave): No aplica (consulta simple filtrada por múltiples campos).
- Observaciones (ej. particiones, índices):
 - Se realiza una conversión de moneda: los montos en USD se multiplican por 4 para equipararse a PEN.

- El valor total obtenido se asigna al campo servicio_dhl.
- La consulta es ejecutada mediante el endpoint method/send-query-database usando \$this->utils->erp("POST", ...).

3.3 Conexión / Base: CAPACITACION

- Tabla / Recurso: tabSalary Slip
- Alias (si aplica): No aplica
- Campos utilizados (clave → descripción):
 - gross_pay → Total bruto pagado en la planilla.
 - name → Identificador del registro de planilla.
 - department → Departamento responsable (filtrado por “Proyectos - SE”).
 - area_del_trabajador → Área asignada del trabajador (filtrado por “Aereo internacional - SE”).
 - start_date → Fecha de inicio del periodo de pago.
- Condiciones (WHERE) aplicadas:
 - start_date = {first_date}
 - department = 'Proyectos - SE'
 - area_del_trabajador = 'Aereo internacional - SE'
 - Agrupación por departamento (GROUP BY department).
- Tipo de unión con otras fuentes (JOIN y llave): No aplica (consulta directa a una tabla).
- Observaciones (ej. particiones, índices):
 - Los resultados agregan:
 - SUM(gross_pay) → monto total pagado (planilla).
 - COUNT(name) → cantidad de empleados (personas).
 - Este valor se usa para calcular los costos de personal asociados al servicio DHL.

3.4 Conexión / Base: REPORTES

- Tabla / Recurso: report_centro_costo_dhl_2_YYYY_MM
- Alias (si aplica): No aplica
- Campos utilizados (clave → descripción):
 - periodo → Mes del reporte (en español).
 - qty_envios → Cantidad total de envíos DHL.
 - ventas → Monto total de ventas DHL.

- ventas_90_porcentaje → Proyección del 90% de las ventas.
 - compras_abastecimiento → Compras totales de abastecimiento (constante 0).
 - servicio_dhl → Monto de servicio DHL cargado desde ERP.
 - planilla → Total de planilla asociado.
 - rentabilidad → Diferencia entre ingresos y egresos.
 - rentabilidad_porcentaje → Porcentaje de rentabilidad.
 - year → Año procesado.
 - created_date, status → Metadatos del registro.
- Condiciones (WHERE) aplicadas:
 - Generación dinámica por mes usando {first_date} y {second_date} del rango procesado.
- Tipo de unión con otras fuentes (JOIN y llave): No aplica (carga por inserción de datos preprocesados).
- Observaciones (ej. particiones, índices):
 - Se crean tablas mensuales dinámicamente (CREATE TABLE report_centro_costo_dhl_2_YYYY_MM).
 - Se aplica un índice adicional: ALTER TABLE ADD INDEX status (status);
 - Se registran logs de ejecución en emp_reportes_validation y error_reports.

4) Filtros globales del reporte

- Inclusiones (must-have):
 - Solo registros con eliminado = 0
 - Solo proveedor "DHL EXPRESS PERÚ S.A.C."
- Exclusiones (reglas de descarte): Registros sin monto (monto <= 0)
- Reglas por estado / habilitado:
 - status inicial = 1 (en tabla destino)
 - Insert en emp_reportes_validation con status = 1 si proceso exitoso
- Parámetros externos (si el usuario puede filtrarlo):
 - Fechas generadas dinámicamente por range_days()

- No recibe input manual del usuario

5) Transformaciones y Reglas de negocio

- Derivaciones de campos (cómo se calculan):
 - $\text{ventas_90_porcentaje} = \text{ventas} * 0.9$
 - $\text{rentabilidad} = \text{ventas_90_porcentaje} - \text{servicio_dhl} - \text{planilla}$
 - $\text{rentabilidad_porcentaje} = (\text{rentabilidad} / \text{ventas_90_porcentaje}) * 100$
- Mapeos de estado: status → 1 (activo)
- Reglas de validación (p.ej., sólo registros validados):
 - Si `obtain_amount_dhl["status"] == false`, el proceso se detiene
 - Reglas condicionales (si X entonces Y): Si $\text{rentabilidad} == 0 \rightarrow \text{rentabilidad_porcentaje} = '0\%'$

6) Estructura de salida

- Tabla/archivo destino: `report_centro_costo_dhl_2_YYYY_MM`
- Particionado / sufijo: Mensual (`_YYYY_MM`)
- Clave(s) primaria(s) o únicas: `id` (AUTO_INCREMENT)
- Columnas del reporte (nombre → tipo → descripción):
 - `id` → INT → Identificador único del registro (PK)
 - `agencia` → VARCHAR(255) → Nombre de la tienda o sucursal
 - `producto_servicio` → ENUM → Clasificación del ítem (producto / servicio)
 - `nombre_articulo` → VARCHAR(225) → Descripción o nombre completo del artículo
 - `stock_inicial` → FLOAT → Cantidad disponible al inicio del periodo
 - `unidades_recibidas` → FLOAT → Total de unidades ingresadas durante el periodo
 - `unidades_vendidas` → FLOAT → Total de unidades vendidas o despachadas
 - `merma` → VARCHAR(20) → Cantidad perdida o deteriorada
 - `inventario_final` → FLOAT → Stock final calculado ($\text{stock_inicial} + \text{recibidas} - \text{vendidas} - \text{merma}$)
 - `costo_de_compra` → VARCHAR(20) → Costo unitario de adquisición
 - `precio_venta` → FLOAT → Precio unitario de venta
 - `valor_inventario_final` → FLOAT → Valor monetario del inventario final
 - `valor_venta` → FLOAT → Ingreso total generado por ventas

- valor_compra → FLOAT → Costo total de adquisición
- rentabilidad → FLOAT → Diferencia entre valor_venta y valor_compra
- ano → INT → Año correspondiente al periodo
- periodo → VARCHAR(20) → Nombre del mes o periodo (ej. "ENERO")
- created_date → DATETIME → Fecha y hora de creación del registro
- status → TINYINT → Estado lógico del registro (1=activo, 0=inactivo)

- Ordenamiento por defecto: ORDER BY year, periodo ASC

7) Frecuencia y operación

- Frecuencia de actualización: PENDIENTE (Esperando info de Eduardo)
- Ventana que recalcula (días/meses): 24 meses
- Tamaño de lote / paginado (chunking): 900 registros por inserción (array_chunk)
- Tolerancia a fallos / reintentos: Transacciones con beginTransaction() y rollBack() ante error
- Tiempos de ejecución esperados: PENDIENTE (Esperando info de Eduardo)

8) Calidad y controles

- Validaciones previas/post:
 - Verificación de existencia de tabla (exist_table_bd)
 - Inserción de log en emp_reportes_validation
- Controles de duplicados: Implícitos por tabla nueva mensual (_YYYY_MM)
- Métricas de completitud (qué se monitorea):
 - Conteo de registros insertados
 - Estado de ejecución (status)

9) Dependencias externas

- APIs / servicios y endpoints: APICAPACITACION/method/send-query-database (ERPNext)

- Autenticación / headers: Controlados por `$this->utils->erp()` (maneja tokens internos)
- Mapeos y contratos de datos: Recibe y envía datos JSON (filters, where, sql_query, tables)

10) Historial de cambios

2025-10-21 18:15 - Elian Franco Arroyo Rodas - Documentación inicial

Revisión #1

Creado 2025-10-24 15:36:27 -05 por Elian

Actualizado 2025-10-24 15:36:35 -05 por Elian