

REPORTE ABASTECIMIENTO

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

1) Nombre del reporte

- Nombre: REPORTE ABASTECIMIENTO
- Código / Alias: report_abastecimiento
- Responsable: Equipo de Reportes Cloud
- Propósito (1 línea):

2) Alcance temporal

- Campo(s) de fecha utilizados: first_date, second_date (obtenidos de obtainRangeDaysByYear)
- Tipo de rango (diario / mensual / personalizado): Mensual — el método obtiene los rangos de días por mes dentro del año actual (\$year, \$last_month, \$first_month).
- Inclusión de horas (sí/no).: Si [00:00:00 – 23:59:59]
- Zona horaria: La del servidor

3) Origen de datos (tablas/APIs)

- Conexión / Base: CDTALLERES
- Tabla / Recurso: report_abastecimiento
- Alias (si aplica): no aplica
- Campos utilizados (clave → descripción):
 - id → Identificador autoincremental
 - name → Nombre del producto o ítem
 - item_group → Grupo o categoría del ítem
 - item_name → Descripción completa del ítem
 - warehouse → Almacén o bodega de registro
 - stock → Cantidad en stock disponible
 - month-entry-0 → Entradas del mes actual

- month-entry-1 → Entradas del mes anterior (1 mes atrás)
 - month-entry-2 → Entradas de hace 2 meses
 - month-entry-3 → Entradas de hace 3 meses
 - month-out-0 → Salidas del mes actual
 - month-out-1 → Salidas del mes anterior (1 mes atrás)
 - month-out-2 → Salidas de hace 2 meses
 - month-out-3 → Salidas de hace 3 meses
 - productValueProm → Valor promedio del producto
 - valueLastBuy → Valor de la última compra
 - lastInspection → Fecha de última inspección del ítem
 - daysInspection → Días transcurridos desde la última inspección
 - cantidad_supervisiones → Número de supervisiones realizadas
 - cantidad_reconciliaciones → Número de conciliaciones registradas
 - sucursal → Sucursal a la que pertenece el almacén
 - departamento → Departamento asociado a la sucursal
 - id_terminal → Identificador del terminal o punto de control
 - created_date → Fecha de creación del registro
 - status → Estado del registro (1=activo, 0=inactivo)
- Condiciones (WHERE) aplicadas:
 - El proceso obtiene datos desde el servicio report() filtrando los registros cuya fecha de creación o actualización se encuentre dentro del rango entre first_day y last_day de cada mes generado en el período actual.
- Tipo de unión con otras fuentes (JOIN y llave):
 - No aplica (los datos se insertan directamente en una tabla creada dinámicamente, sin uniones con otras tablas).
- Observaciones (ej. particiones, índices):
 - Se crea un índice sobre el campo status mediante ALTER TABLE para optimizar las consultas.
 - Las tablas se generan dinámicamente según el rango de fechas (CREATE TABLE report_abastecimiento ...) y se validan en emp_reportes_validation.

4) Filtros globales del reporte

- Inclusiones (must-have): Todos los registros generados dentro del rango mensual definido.

- Exclusiones (reglas de descarte): No se definen exclusiones explícitas.
- Reglas por estado / habilitado: Solo se indexa y marca con status = 1 por defecto.
- Parámetros externos (si el usuario puede filtrarlo): No recibe filtros desde el usuario; todo es generado internamente.

5) Transformaciones y Reglas de negocio

- Derivaciones de campos (cómo se calculan): Ninguna derivación directa; los datos se insertan tal como los devuelve `$this->report()`.
- Mapeos de estado:
 - status = 1 → ejecución exitosa,
 - status = 0 → error durante el proceso.
- Reglas de validación (p.ej., sólo registros validados):
 - Se valida la existencia de la tabla con `verifyExistTable`.
 - Si no existe o hay error, se registra en `error_reports`.
- Reglas condicionales (si X entonces Y):
 - Si `verifyExistTable` retorna false, se salta la carga y se guarda error.
 - Si hay excepción, se registra como error en `emp_reportes_validation`.

6) Estructura de salida

- Tabla/archivo destino: `report_abastecimiento`
- Particionado / sufijo: No se define; tabla única.
- Clave(s) primaria(s) o únicas: `id` (PRIMARY KEY)
- Columnas del reporte (nombre → tipo → descripción):
 - `id` → INT → Identificador autoincremental del registro
 - `name` → VARCHAR(255) → Nombre del producto
 - `item_group` → VARCHAR(150) → Grupo o categoría del ítem
 - `item_name` → VARCHAR(255) → Nombre completo del ítem
 - `warehouse` → VARCHAR(150) → Almacén de procedencia
 - `stock` → DOUBLE → Stock actual disponible
 - `month-entry-0..3` → DOUBLE → Entradas por mes (últimos 4 meses)
 - `month-out-0..3` → DOUBLE → Salidas por mes (últimos 4 meses)
 - `productValueProm` → DOUBLE → Valor promedio del producto

- valueLastBuy → DOUBLE → Valor de la última compra registrada
- lastInspection → DATE → Fecha de la última inspección realizada
- daysInspection → INT → Días transcurridos desde la última inspección
- cantidad_supervisiones → INT → Total de supervisiones registradas
- cantidad_reconciliaciones → INT → Total de conciliaciones realizadas
- sucursal → VARCHAR(150) → Sucursal de registro
- departamento → VARCHAR(150) → Departamento asociado
- id_terminal → VARCHAR(100) → Identificador del terminal
- created

- Ordenamiento por defecto: No se define explícitamente.

7) Frecuencia y operación

- Frecuencia de actualización: PENDIENTE (Esperando info de Eduardo)
- Ventana que recalcula (días/meses): Un mes por ejecución.
- Tamaño de lote / paginado (chunking): Inserciones en lotes de 300 registros (array_chunk(\$data_report, 300)).
- Tolerancia a fallos / reintentos: Control mediante try/catch con registro en tabla de errores.
- Tiempos de ejecución esperados: PENDIENTE (Esperando info de Eduardo)

8) Calidad y controles

- Validaciones previas/post:
 - Verificación de existencia de tabla antes de inserción.
 - Registro de validación en emp_reportes_validation.
- Controles de duplicados: No se implementan explícitamente.
- Métricas de completitud (qué se monitorea): No se definen métricas automáticas; sólo logs de éxito/error.

9) Dependencias externas

- APIs / servicios y endpoints: Interno → `$this->reportcontroller->obtainRangeDaysByYear()`, `verifyExistTable()`, `consultaSql()`
- Autenticación / headers: No aplica (llamadas internas).
- Mapeos y contratos de datos: `$this->report()` devuelve el dataset principal que se inserta en la tabla.

10) Historial de cambios

2025-10-18 12:53 - Elian Franco Arroyo Rodas - Documentación inicial

Revisión #1

Creado 2025-10-24 15:34:23 -05 por Elian

Actualizado 2025-10-24 15:34:33 -05 por Elian