

control/grupo/listar - [index] (1,2)

Path: /var/www/html/qaservicesapp/app/Http/Controllers/ShalomControl/GrupoController.php

? Descripción

Retorna una lista paginada de **grupos de cargueros de desembarque**, filtrados por terminal, nombre y fecha.

Cada grupo incluye sus placas asociadas y el nombre legible de la terminal.

Realiza consultas en la base **empresarial**, transforma datos y enriquece la respuesta con nombres de terminales y placas de cargueros programados.

? Endpoint

POST /cargueros-desembarque

? Request Body

```
{
  "terminal": 3,
  "pagina": 1,
  "nombre": "TURNO A"
}
```

Parámetros

Campo	Tipo	Obligatorio	Descripción
terminal	int	✓ Sí	ID de la terminal.
pagina	int	✓ Sí	Número de página (paginación de 15 registros).

Campo	Tipo	Obligatorio	Descripción
nombre	string	✘ Opcional	Filtra por nombre del grupo (activa rango de fecha extendido).

? Seguridad

Requiere token o autenticación definida en middleware general (no se valida dentro del método).

? Flujo del Servicio (resumen real)

1. Validaciones iniciales

- Si no envían `terminal` → error obligatorio.
- Se toma `pagina` para paginar.
- Se define `limit = 15`.
- Fecha mínima:
 - Por defecto: **ayer**
 - Si se envía `nombre`: se amplía rango **-10 días a +10 días** y se agrega filtro LIKE al nombre.

2. Construcción del filtro para la consulta

Filtros aplicados:

- `terminal = enviado`
- `status = 1`
- `fecha_creacion > fechaDesde`
- si existe nombre → LIKE `%nombre%` y `fecha_creacion < fechaHasta`

3. Consulta de grupos de desembarque

Tabla: `emp_grupo_cargueros_desembarque`

Campos retornados:

- id
- grupo (array JSON de cap_id)
- nombre
- terminal
- cerrar_grupo
- fecha_creacion
- placas = null (placeholder)

Si no hay resultados → “No se encontraron grupos”.

4. Obtención del catálogo de terminales

Consulta `emp_terminal` y crea un arreglo tipo diccionario:

```
[ ter_id => ter_nombre ]
```

5. Enriquecimiento de la data

Para cada grupo:

1. Decodifica `grupo` (lista de cap_id).
 2. Consulta placas desde:
 - **emp_cargprogramado (cap)**
 - **emp_carguero_new (carg)**Campos:
 - placa1
 - codigo1
 - placa2
 3. Construye lista de placas:
`"placa1 codigo1"`
 4. Reemplaza el id de terminal por su nombre real.
-

? Response 200 – Ejemplo

```
{
  "success": true,
  "message": "Lista de grupos",
  "data": [
    {
      "id": 15,
      "grupo": [123, 124],
      "nombre": "TURNO A",
      "terminal": "Terminal Callao",
      "cerrar": 0,
      "fecha": "2025-01-26 10:21:00",
      "placas": [
        "B1234 C001",
        "B2288 C002"
      ]
    }
  ]
}
```

? Posibles Errores

1. Terminal obligatorio

```
{
  "success": false,
  "message": "Es obligatorio enviar una terminal"
}
```

2. Sin resultados

```
{
  "success": false,
  "message": "No se encontraron grupos"
}
```

? Schemas (estructuras usadas)

emp_grupo_cargueros_desembarque

```
{  
  "id": "number",  
  "grupo": "json array of cap_id",  
  "nombre": "string",  
  "terminal": "number",  
  "cerrar_grupo": "number",  
  "fecha_creacion": "datetime"  
}
```

emp_cargprogramado + emp_carguero_new

```
{  
  "placa1": "string",  
  "codigo1": "string",  
  "placa2": "string"  
}
```

emp_terminal

```
{  
  "ter_id": "number",  
  "ter_nombre": "string",  
  "ter_abrebiatura": "string"  
}
```

? Lógica en pseudo-código

```
if !terminal -> error

fechaDesde = yesterday
if nombre:
  fechaDesde = -10 days
  fechaHasta = +10 days
  add filter nombre LIKE %nombre%
  add filter fecha_creacion < fechaHasta

add filter fecha_creacion > fechaDesde

grupos = SELECT from emp_grupo_cargueros_desembarque where filters, paginated
if empty -> error

terminales = SELECT all terminales
map[ter_id] = ter_nombre

for each grupo:
  ids = decode(grupo.grupo)
  cargueros = SELECT placas info using ids
  grupo.placas = format placas
  grupo.terminal = terminales[grupo.terminal]

return grupos
```

Revisión #2

Creado 2025-11-27 15:30:02 -05 por Elian

Actualizado 2025-11-27 16:01:41 -05 por Elian